



Etude du comportement humain grâce à la simulation multi-agents et aux méthodes de fouille de données temporelles

Stephane Daviet

► To cite this version:

Stephane Daviet. Etude du comportement humain grâce à la simulation multi-agents et aux méthodes de fouille de données temporelles. Informatique [cs]. Université de Nantes, 2009. Français. NNT : . tel-00482642

HAL Id: tel-00482642

<https://theses.hal.science/tel-00482642>

Submitted on 11 May 2010

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Étude du comportement humain grâce à la simulation multi-agents et aux méthodes de fouille de données temporelles

Mémoire de Thèse

présentée et soutenue publiquement le vendredi 13 mars 2009

pour l'obtention du grade de

Docteur de l'Université de Nantes

Spécialité Informatique

par

Stéphane Daviet

Composition du jury

Rapporteurs : Suzanne Pinson, Professeur, Université Paris-Dauphine
Djamel Zighed, Professeur, Université de Lyon 2

Examineurs : Stefan Trausan-Matu, Professeur, Université polytechnique de Bucarest
Henri Briand, Professeur, École polytechnique de l'Université de Nantes
Pascale Kuntz, Professeur, École polytechnique de l'Université de Nantes

Directeurs : Fabrice Guillet, Maître de Conférence, HdR, École polytechnique de l'Université de Nantes

Mis en page avec la classe thloria.

Remerciements

Je remercie bien évidemment Fabrice Guillet, co-encadrant puis directeur de cette thèse, qui m'a accompagné, de près comme de loin, à Nantes ou à Niort et qui m'a fait bénéficier de son expérience du monde de la recherche. Je remercie également Henri Briand, directeur initial de cette thèse, sous l'impulsion duquel j'ai pu mener cette thèse.

Je remercie particulièrement Suzanne Pinson et Djamel Zighed pour avoir accepté d'être les rapporteurs de cette thèse, ainsi que Stefan Trausan Matu, Henri Briand, Pascale Kuntz de faire partie des examinateurs.

Pascale Kuntz, directrice de l'équipe COD, a également contribué d'une manière importante à ces travaux par des conseils avisés et une aide toujours bienvenue. Mes remerciements iront également à Jacques et Vincent Philippé sans qui le sujet de cette thèse n'existerait sans doute pas et desquels j'ai pu apprendre sur la conduite d'une thèse en entreprise. Ces remerciements sont également adressés à la société PerformanSe qui a concouru à la réalisation de ces travaux. Merci également à APARTA à l'origine de ce sujet de recherche.

Cette thèse CIFRE s'étant déroulé à la fois au labo et en entreprise, mes remerciements s'adressent aussi aux nombreuses personnes que j'ai pu y côtoyer. Au labo d'abord, Marie-Pierre qui a toujours su faire preuve d'encouragements bienvenus, Catherine pour la pertinence de ses traductions, Julien pour sa gentillesse, sa disponibilité et son sérieux, Nicolas dont j'ai apprécié la fantaisie. À KnoweSia et PerformanSe ensuite, Serge Baquedano dont l'expertise dans le domaine de la psychologie a été précieuse, Francis avec lequel j'ai apprécié travailler, Tudor et toute l'équipe de KnoweSia pour l'aventure très enrichissante que j'ai partagée avec eux. Avec un pied dans chaque monde, merci à Bruno pour toutes les conversations qu'il a pu initier et à Jérôme pour sa compagnie très agréable.

Merci à Catherine, organisatrice hors paire. À Manu qui a partagé avec moi les turpitudes de thésard, m'a initié au sport avec plus ou moins de réussite et avec lequel il a toujours été passionnant de discuter de mille sujets. À Philippe pour avoir efficacement égayé l'espace de notre bureau. Je remercie également Angélique, Arnaud, Élisabeth, Brice, Vincent, Gaële, (Jean) Arnaud, Farid, Romain, Sabine, Christophe, Annabelle.

Merci à Pierre mon petit bonhomme pour ses petits bonheurs et, pour tout ce qu'elle est et ce qu'elle m'apporte, merci à ma femme, Hélène.

À Hélène et mon petit Pierre.

Table des matières

Table des figures	ix
Liste des tableaux	xi
Introduction	1
Chapitre 1 La modélisation comportementale et émotionnelle d'un individu	7
1.1 Cadre applicatif	8
1.2 Les modèles comportementaux	9
1.2.1 Historique des modèles comportementaux	9
1.2.2 Les outils d'évaluation comportementale	10
1.2.3 Le modèle comportemental PerformanSe	11
1.2.4 Spécificités des personnes cérébrolésées	15
1.3 Le système cognitif humain	16
1.3.1 L'attention	17
1.3.2 La mémoire	18
1.3.3 Le langage	18
1.3.4 La vision et l'imagerie mentale	19
1.3.5 Les fonctions exécutives	20
1.4 Les émotions	21
1.4.1 Définition	21
1.4.2 Théories des émotions	22
1.4.3 Le rôle des émotions	27
1.4.4 Le modèle OCC	28
1.5 Humeurs et tempérament	30
1.6 Les sentiments	30

1.7	L'influence de l'environnement	31
1.7.1	Les facteurs d'ambiance	31
1.7.2	Les interactions sociales	33
Chapitre 2 Les systèmes multi-agents		35
2.1	Principes et concepts	36
2.1.1	Historique	36
2.1.2	Définition	41
2.1.3	L'environnement	49
2.2	Les modèles d'agents	51
2.2.1	La problématique de la modélisation agent	51
2.2.2	L'agent non-délibératif	53
2.2.3	L'agent délibératif	55
2.2.4	Les agents hybrides	57
2.2.5	L'agent émotionnel	58
2.3	Les interactions entre agents	58
2.3.1	Les interactions entre agents	60
2.3.2	La communication entre agents	61
2.3.3	La notion de concurrence	62
2.3.4	Les langages de communication entre agents	63
Chapitre 3 Fouille de données multidimensionnelles		67
3.1	Historique, objectif et principe général de la fouille de données	68
3.2	La simulation et la problématique des données multidimensionnelles	70
3.2.1	Les dimensions spatiale et temporelle	70
3.3	Techniques et outils de stockage de données multidimensionnelles	75
3.3.1	La notion de temps et les logiques temporelles	75
3.3.2	Les logiques temporelles	75
3.3.3	L'intégration du temps et de l'espace dans les SGBD relationnels et objets	76
3.4	Data mining temporel	79
3.4.1	Préliminaires	80
3.4.2	Les tâches classiques d'étude de données temporelles	82
3.4.3	ECD sur des données temporelles : la découverte de motifs	86
3.4.4	La méthode des projections aléatoires	91

3.5	Data mining spatial	93
3.5.1	Les techniques de data-mining spatial	93
3.5.2	Notion de sémantique d'une position et problématique	94
Chapitre 4 Un nouveau modèle d'agent émotionnel : l'agent EFT		97
4.1	Un modèle d'agent émotionnel : l'agent EFT	98
4.1.1	Structure générale de l'agent	98
4.1.2	Processus de contrôle de l'agent	110
4.2	Les interactions du système	114
4.2.1	Présentation du système étudié	114
4.2.2	Modélisation de protocoles d'interaction	118
4.2.3	Modélisation d'interactions spécifiques	120
Chapitre 5 Un modèle de fouille de trace d'agents		127
5.1	Données issues de la simulation	128
5.1.1	Des données propres	128
5.1.2	Des données malléables	128
5.1.3	Des données à volonté	129
5.2	Pré-traitement des données	129
5.2.1	Discrétisation des données	129
5.2.2	Dissimilarité entre vecteurs	130
5.2.3	Réduction dimensionnelle	137
5.2.4	Agrégation de vecteurs	137
5.3	Méthode des projections aléatoires	141
5.3.1	Présentation générale de la méthode	141
5.3.2	Étapes du traitement	141
5.3.3	Mise en oeuvre de la méthode	143
Chapitre 6 Implémentation de la simulation et résultats obtenus		151
6.1	Implémentation de la plate-forme de simulation	152
6.1.1	Le langage Java	152
6.1.2	La plate-forme SMA JADE	152
6.1.3	Le logiciel R	153
6.2	Déroulement de la simulation	154

Table des matières

6.3 Fouille des traces d'agent	156
Conclusion	161
Bibliographie	165
Annexes	183

Table des figures

1	Présentation du processus de valorisation des données de simulation.	3
1.1	Exemple de question ipsative du logiciel Echo.	13
1.2	Profil bipolaire Echo sur 10 dimensions.	13
1.3	Système Echo à base de règles.	16
1.4	Modèle de contrôle attentionnel de Norman and Shallice (1980).	21
1.5	Expressions universelles d'Ekman.	25
1.6	Théorie des émotions de Schachter and Singer (1962).	25
1.7	Théorie des émotions de Arnold (1950).	26
1.8	Structure globale des types d'émotions (Ortony et al., 1988).	29
2.1	Agent réactif.	53
2.2	Agent BDI.	57
2.3	Agents hybrides (architectures en couches).	59
3.1	Problématique de l'analyse des données de simulation	71
3.2	Structures temporelles a) linéaire, b) et c) ramifiée et d) cyclique.	73
3.3	Structures temporelles pour 1 à n simulations.	74
3.4	Datation par points et par intervalles.	76
3.5	Décomposition d'une série chronologique.. . . .	81
4.1	Modèle UML de l'agent EFT.	99
4.2	Caractéristiques personnelles de l'agent.	100
4.3	Variabilité des différents composants de l'affectif humain.	101
4.4	Structure de la mémoire.	106
4.5	Processus de contrôle de l'agent.	112
4.6	Chaîne de conditionnement.	115
4.7	Structures relationnelles entre agents.	116
4.8	Diagramme de cas d'utilisation global.	117
4.9	Diagramme de cas d'utilisation du poste 9.	117
4.10	Protocole de perception volontaire.	118
4.11	Protocole d'interaction avec le poste de travail.	119
4.12	Coefficient d'intention d'un message dialogue.	120
4.13	Protocole de remarque entre individus.	121

Table des figures

4.14	Protocole de demande d'aide entre individus.	122
4.15	Diagramme d'activités du fonctionnement normal de la chaîne.	123
4.16	Diagramme d'activités du poste 7.	124
5.1	Illustration de la procédure de sous-échantillonnage.	131
5.2	Matrice de dissimilarité sur les données de la figure 5.1.	136
5.3	Illustration de la procédure de réduction dimensionnelle.	138
5.4	Principe d'utilisation des projections aléatoires dans Duchêne (2004) pour la tolérance au bruit.	143
6.1	Capture d'écran de la fenêtre principale de la simulation.	155
6.2	Représentation des émotions.	155
6.3	Évolution de l'état interne de l'agent.	156
6.4	Capture des données réduites dans R.	157
6.5	Support des sous-séquences de taille 4 pour le masque X^*XX	157
6.6	Support des sous-séquences de taille 4 pour le masque XX^*X	158
6.7	Support des sous-séquences de taille 4 pour le masque $X^{**}X$	158
6.8	Support des sous-séquences de taille 4 pour le masque $XXXX$	158

Liste des tableaux

1.1	Les 10 dimensions comportementales.	14
1.2	Liste des émotions fondamentales proposées par chaque auteur (Christophe, 1998).	24
1.3	Température au travail selon Grandjean (1985).	32
2.1	Taxonomie des agents.	52
2.2	Performatives KQML.	63
2.3	Attributs des messages KQML.	64
2.4	Performatives FIPA ACL (Wooldridge, 2001).	64
3.1	Propriétés des SGBDs temporels avec datation par points et par intervalles. . .	75
3.2	Taxonomie des logiques temporelles (von Bertalanffy, 1973).	76
3.3	Intégration des historiques dans le modèle relationnel.	78
4.1	Caractéristiques psychologiques de l'agent.	104
4.2	Formalisation des familles de règles (familles de 1 à 10).	109
1	Les émotions de bien-être.	184
2	Les émotions liées aux ressentis pour les autres.	184
3	Les émotions liées à un évènement éventuel.	184
4	Les émotions d'attribution.	185
5	Les émotions composées : bien-être et attribution.	185
6	Les émotions d'interaction.	185

Introduction

La simulation comportementale

Certains systèmes réels, au comportement ou à la structure complexe, échappent aux outils d'étude classiques. L'observateur n'a alors pas les moyens de saisir l'ensemble des tenants et aboutissants de ces systèmes caractérisés par un grand nombre d'interactions internes et externes, une structure non décomposable et un comportement imprédictible. L'étude des comportements sociaux d'un groupe d'individus s'inscrit dans cette problématique. Ces dernières années, l'informatique a néanmoins permis de développer des outils de simulation de plus en plus performants permettant notamment le traçabilité des phénomènes, la réitération des expériences, la capture de moments clés et la possibilité d'étude de cas inexistantes ou encore jamais survenus en milieu réel. À partir d'une modélisation pertinente du système, on parvient aujourd'hui à simuler son évolution de plus en plus fidèlement. Au niveau de l'étude de l'individu et des interactions sociales, les sciences humaines apportent également leur expertise dans ce processus de modélisation. À la confluence de ces deux disciplines, on entrevoit aujourd'hui la possibilité de modéliser et simuler le comportement humain.

Les études dans le domaine des modèles comportementaux se sont particulièrement développées au cours du XX^{ème} siècle. Les progrès conjugués de la psychologie, de la médecine et des sciences cognitives ont permis de croiser des données expérimentales avec les modèles proposés par les chercheurs. On peut distinguer deux axes d'études : l'étude du caractère de l'individu et celle des émotions ressenties. Pour le premier, les modèles de traits comportementaux tel que le *Big Five* (Goldberg, 1990) ont connu un certain succès, renforcé dans les 20 dernières années par le développement du secteur des ressources humaines et de l'outil informatique. Pour le second, on a assisté à la démonstration de l'importance des émotions dans le processus décisionnel humain et à l'établissement d'une catégorisation largement adoptée des émotions principales de l'être humain dans des modèles tels qu'OCC.

D'autre part, les travaux effectués conjointement dans les domaines de l'intelligence artificielle et de la vie artificielle ont fourni matière à l'apparition du domaine des systèmes multi-agents. Ce domaine de recherche, axé sur la création d'entités virtuelles autonomes, a connu de grandes avancées dans les 20 dernières années avec l'apparition de modèles d'agent reconnus tel que le modèle *BDI* (Bratman, 1987; Rao and Georgeff, 1991). Les SMAs sont aujourd'hui utilisés dans des domaines très variés : des robots explorateurs martiens aux simulations de

phénomènes biologiques. Les avancées des modèles comportementaux et des SMAs laissent aujourd'hui entrevoir la possibilité de simuler le comportement humain.

La fouille de données complexes

La simulation offre de grandes facilités d'étude de phénomènes parfois complexes à observer ou difficilement reproductibles, mais présente l'inconvénient de produire un important volume de données. Ainsi, il peut parfois être difficile en ayant recours à une simulation d'observer précisément un phénomène ou d'avoir une vision synthétique de son déroulement. La fouille de données et plus généralement l'ECD ont spécifiquement traité ces problématiques d'extraction de connaissances pertinentes à partir d'un important volume de données. Il s'agit d'appliquer des algorithmes et des méthodes afin de trouver des éléments congrus pour l'utilisateur parmi ces données. À la différence des méthodes statistiques couramment employées pour étudier d'importants volumes de données et qui produisent des résultats et des indicateurs globaux sur leur comportement général, les méthodes d'ECD recherchent parmi les données celles qui individuellement ou associées sont les plus intéressantes pour l'utilisateur. Ces résultats sont généralement issus d'une recherche en profondeur la plupart du temps à l'aide d'un processus itératif où l'implication de l'utilisateur et la liberté du processus de fouille peuvent être variable. On distingue les méthodes dites *supervisées*, *non-supervisées* et *semi-supervisées*. Dans les méthodes supervisées, l'objectif est de déterminer le lien entre variables explicatives — exogènes, et variables à expliquer — endogènes. À l'inverse, dans les méthodes non supervisées, les résultats de la fouille sont automatiquement extraits des données sans présupposer du rôle exogène ou endogène de chaque variable.

Les travaux d'Agrawal and Imielinski (1993) sur la méthode *a priori* permettant l'extraction de motifs fréquents à partir d'une base de données *individu* \times *variables* sont considérés comme initiateurs de l'extraction de connaissances à partir des données. Cette recherche de motifs fréquents reste aujourd'hui un sujet de recherche ouvert, principalement du fait de la diversité des types de données à étudier. Avec l'avènement de l'ère d'Internet, la multiplication des sources de données et du volume global de ces données pose continuellement de nouveaux défis. Il devient notamment très intéressant d'étudier l'évolution des habitudes des utilisateurs dans le temps. La fouille de données séquentielles traite spécifiquement cette problématique. Ainsi, des méthodes d'ECD spécialisées dans la recherche de motifs fréquents sur des données temporelles se sont développées. La méthode FACE (Dousson and Duong, 1999) par exemple propose une solution pour découvrir en temps réel dans un flux de journaux d'alarmes réseau des séquences caractéristiques. Duchêne (2004) a, elle, proposé une adaptation de la technique des projections aléatoires pour l'étude de données issues de la surveillance médicale à domicile. Dans le cadre de l'étude de données issues d'une simulation, ces méthodes de fouille peuvent s'avérer particulièrement adaptées.

Contexte applicatif

Nos travaux de recherche ont été conduits dans le cadre d'une convention CIFRE entre l'équipe COD du LINA et la société PerformanSe. Celle-ci, basée à Carquefou, est spécialisée dans les outils d'évaluation comportementale et d'accompagnement de l'individu dans son parcours professionnel. Ces outils sont créés à destination des professionnels des ressources humaines. Le logiciel Dialecho permet à un individu de s'auto-évaluer en répondant à 70 questions portant sur son comportement au travail. L'outil établit alors le profil comportemental de l'individu selon 10 dimensions bipolaires. Un ensemble d'outils complémentaires offrent d'autres axes d'étude : activités dans le travail, perception de soi, perception de soi par les autres, etc.

Disposant principalement d'outils destinés à établir une capture à un instant donné du profil comportemental d'un individu, PerformanSe souhaitait étudier la faisabilité d'une simulation comportementale pour observer l'évolution de ce comportement dans le temps. Plus particulièrement, l'objectif était, dans un premier temps, d'appliquer cette méthodologie à l'étude du comportement de personnes cérébrolésées dans leur milieu professionnel dans un centre d'aide par le travail, ceci afin de contribuer à leur réinsertion dans le milieu professionnel ordinaire. L'outil d'aide à la décision ainsi conçu peut ensuite être étendu à d'autres domaines professionnels. Nos travaux de recherche visent donc à produire un outil de simulation comportementale mettant en scène des individus dans leur environnement de travail. Nous souhaitons ensuite découvrir par la fouille des données produites, des séquences d'états comportementaux caractéristiques d'une situation. La figure illustre le processus mis en œuvre.

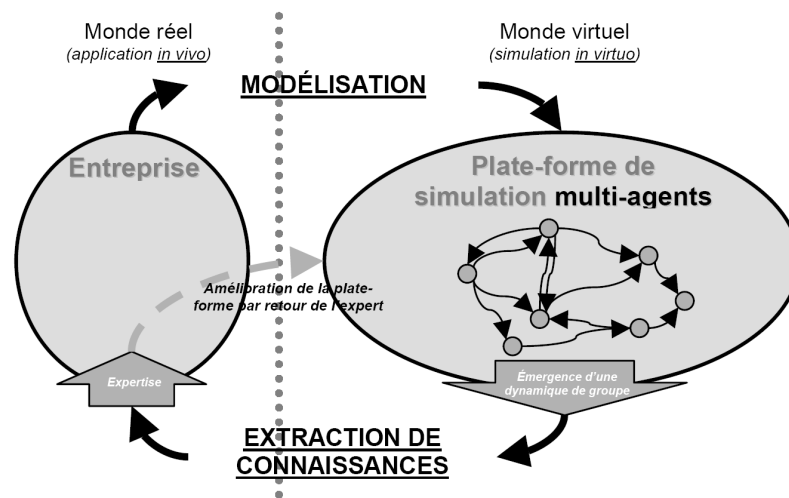


FIG. 1 – Présentation du processus de valorisation des données de simulation.

Contributions de la thèse

Nos travaux se déclinent en trois thèmes principaux. Nous proposons, d’une part, un modèle d’agent émotionnel original : l’agent EFT. L’agent cognitif le plus utilisé dans la littérature est l’agent *BDI* (*Belief, Desire, Intention*) qui fournit un modèle efficace du processus de délibération humaine. Il ne comporte néanmoins aucune dimension affective. Nous enrichissons ce modèle en ajoutant une partie affective influant sur les décisions de l’agent. Nous déclinons la partie affective en trois dimensions complémentaires : les émotions, les sentiments et le tempérament. Les modèles d’agents émotionnels existants ne se focalisent que sur les émotions. Nous considérons pour notre part que les émotions ne décrivent qu’un ressenti fugace. C’est pourquoi nous proposons de prendre également en compte la notion de sentiments qui se construisent à partir des émotions mais dont la variabilité est sur du moyen terme, et la notion de tempérament décrivant le caractère général et quasi immuable sur le long terme de l’individu. Ce modèle en 3 couches influe à la fois le comportement interne de l’agent et ses relations avec l’extérieur. Nous proposons donc un modèle décrit en AgentUML couvrant l’ensemble de ces aspects.

Nous mettons ensuite en pratique cette modélisation par la réalisation d’une plate-forme de simulation multi-agents. Pour ce faire, nous nous basons sur des systèmes de conception agent. Nous avons ainsi utilisé la plate-forme JADE (Java Agent DEvelopment framework) comme socle de nos développements. L’observation et l’interprétation critique de multiples exécutions de la simulation ont contribué à l’ajustement du paramétrage et à l’affinement des comportements observables.

Nous développons enfin une méthode d’étude des données produites par la simulation. Nous disposons de l’évolution de l’ensemble des variables comportementales des agents de la simulation. Nous cherchons à étudier ces séquences temporelles pour y découvrir des motifs fréquents de succession d’événements. Pour ce faire, nous proposons une adaptation de la méthode des projections aléatoires. Nos travaux concernent à la fois les phases de pré-traitement des données et de fouille proprement dite. Pour le pré-traitement, nous proposons une abstraction des données via les étapes classiques de discrétisation, de réduction dimensionnelle et d’agrégation d’événements consécutifs similaires. Pour la fouille, après avoir effectué un découpage de la séquence temporelle recueillie en sous-séquences de taille définie, nous adaptons l’utilisation originale des masques de projections pour découvrir les enchaînements fréquents d’événements consécutifs ou non. Nous proposons une analyse des résultats obtenus en regard du paramétrage et du déroulement observé de la simulation.

Organisation du document

Ce mémoire s’articule autour de deux parties principales : l’état de l’art des domaines concernés et les apports des travaux de recherches ici présentés. Ces deux parties se déclinent chacune en trois chapitres.

Dans la première partie, le **chapitre 1** dresse un bilan des travaux de recherche dans le domaine de la modélisation comportementale, du tempérament de l’individu à ses émotions.

Nous y détaillons les modèles utilisables dans le cadre d'une modélisation informatique du comportement humain. Ce chapitre traite les modèles purement comportementaux décrivant les grands traits de caractère de l'individu, et plus spécifiquement le modèle des *Big Five*, les modèles émotionnels et plus particulièrement le modèle OCC, ainsi que les aspects sociaux d'interactions entre individus et entre un individu et son environnement physique. Le **chapitre 2** présente les systèmes multi-agents¹, domaine qui s'intéresse à la création d'entités virtuelles autonomes. Y sont abordés l'historique du domaine, les définitions et modèles d'agent et les aspects interactionnels des systèmes multi-agents. Le **chapitre 3** dresse l'état de l'art de la fouille de données complexes. Le principe de l'extraction de connaissances à partir de données y est expliqué, ainsi que les techniques de stockage de données complexes, et plus précisément de données spatio-temporelles, et les techniques de fouille de ces données.

Dans la seconde partie, le **chapitre 4** présente les résultats de nos travaux pour ce qui a trait à la simulation du comportement humain. Nous y détaillons notre modèle d'agent émotionnel, l'agent EFT, à la fois pour la partie interne à l'agent et pour les aspects interactions du système, en réalisant la synthèse des notions abordées aux chapitres 1 et 2. Nous verrons dans le **chapitre 5** nos apports dans le domaine de la fouille des données issues de la simulation. Une adaptation de la méthode des projections aléatoires y est présentée. Enfin, le **chapitre 6** traite des résultats à la fois du processus de simulation et de l'étude des données produites par ce dernier.

¹Généralement désignés par l'acronyme *SMA* ou *MAS* pour *Multi Agent Systems* en anglais.

1

La modélisation comportementale et émotionnelle d'un individu

Ce chapitre décrit les modèles comportementaux les plus fréquemment rencontrés en sciences humaines, notamment dans le contexte d'une modélisation de l'Homme au travail.

Sommaire

1.1	Cadre applicatif	8
1.2	Les modèles comportementaux	9
1.2.1	Historique des modèles comportementaux	9
1.2.2	Les outils d'évaluation comportementale	10
1.2.3	Le modèle comportemental PerformanSe	11
1.2.4	Spécificités des personnes cérébrolésées	15
1.3	Le système cognitif humain	16
1.3.1	L'attention	17
1.3.2	La mémoire	18
1.3.3	Le langage	18
1.3.4	La vision et l'imagerie mentale	19
1.3.5	Les fonctions exécutives	20
1.4	Les émotions	21
1.4.1	Définition	21
1.4.2	Théories des émotions	22
1.4.3	Le rôle des émotions	27
1.4.4	Le modèle OCC	28
1.5	Humeurs et tempérament	30
1.6	Les sentiments	30
1.7	L'influence de l'environnement	31
1.7.1	Les facteurs d'ambiance	31
1.7.2	Les interactions sociales	33

Introduction

La modélisation du comportement humain est un sujet qui n'a eu de cesse de susciter l'intérêt des chercheurs tout domaine confondu. Ce fût logiquement dès l'Antiquité un des premiers champs d'investigation de la philosophie qui a très tôt essayé de décortiquer les mécanismes à la base du comportement et plus généralement de la pensée. À la fin du XIX^{ème} siècle, la psychologie a continué dans les traces initiées par les philosophes en essayant cette fois-ci de construire des modèles médicaux et rationnels autour du comportement. C'est réellement avec l'apparition de l'informatique que l'on a vu là une voie pour utiliser ces modèles afin de répliquer avec plus ou moins de succès le raisonnement humain dans les machines.

Dans nos travaux, nous nous sommes inscrits dans cette voie en tentant de doter une entité d'un comportement affectif crédible. Nous verrons ainsi dans cette partie les études sur le comportement humain sur lesquelles nous nous sommes appuyés. Nous présenterons dans un premier temps le cadre applicatif de nos travaux. Nous détaillerons ensuite l'historique des modèles comportementaux et ceux que nous avons choisis pour construire notre entité. Nous verrons ensuite le fonctionnement cognitif de l'individu, puis son système affectif. Nous traiterons enfin de l'influence des facteurs environnementaux sur le comportement de l'individu à la fois pour ce qui a trait à l'environnement naturel de l'individu que pour ce qui concerne ses interactions avec les autres individus.

1.1 Cadre applicatif

À l'origine, le projet « TC&Plus.Virtuel », partenariat entre les entreprises PerformanSe, spécialisée dans les outils d'évaluation comportementale, et Aparta, un atelier protégé visant à la réinsertion par le travail de traumatisés crâniens, avait pour finalité la réalisation d'une plateforme de simulation en réalité virtuelle mettant en scène les employés d'Aparta travaillant sur une de leur machine. La chaîne de production étudiée était une chaîne de conditionnement de paquets de gâteaux par lots (scellage, pose d'étiquettes, mise en carton, ...). Ce premier projet avait donné lieu à une première réalisation accréditant la faisabilité et la pertinence de la démarche. Dans un contexte similaire, nous avons eu pour but de créer cette fois-ci un véritable modèle permettant de simuler le comportement humain, en développant plus particulièrement la partie affective de ce comportement. Cette démarche, à mis chemin entre le monde de l'informatique et celui de la psychologie, a donc nécessité un important travail de recherche sur les modèles comportementaux existants et leur adéquation avec nos besoins.

L'intérêt premier d'une telle simulation est d'en retirer des connaissances sur les postes de travail, sur l'ambiance de l'équipe. Il s'agit donc de substituer à une observation *in vivo* coûteuse en temps, une simulation *in virtuo* de ces comportements individuels et de groupe. Pour atteindre ce but, les agents doivent donc être les plus proches possibles de l'Homme. En effet, pour pouvoir envisager une quelconque mise en valeur des connaissances produites, celles-ci doivent être valides. Il n'est donc pas envisageable de ne pas prendre en compte les émotions et plus généralement le tempérament, parties intégrantes du comportement, rendant le

processus de décision et le raisonnement possibles et les relations humaines riches et variées. La recherche dans les émotions artificielles prend, en général, en compte les émotions à un seul niveau (expression, planification, transmission, ...). Notre objectif a dès le départ afin de rendre la simulation crédible de couvrir le plus largement possible l'éventail de l'influence des émotions sur l'ensemble du comportement humain.

La particularité de notre cadre applicatif réside dans l'étude de personne *cérébrolésées* dans le milieu professionnel. Ces personnes, également qualifiées de *traumatisé crâniens*, ont subi un traumatisme violent à la tête, la plupart du temps dû à un choc². L'affectif en temps normal joue déjà un rôle prépondérant dans les processus de prise de décision comme a pu le démontrer Damasio (1994), néanmoins la plupart des personnes parviennent à réguler leurs émotions dans leur milieu professionnel et à adopter un comportement réfléchi et cartésien. Le traumatisme crânien a pour particularité, de par les lésions subies par les hémisphères cérébraux responsables du comportement raisonné, d'affecter à des degrés divers les filtres émotionnels dont chacun de nous est pourvu. Les personnes cérébrolésées peuvent donc présenter, en plus de troubles psycho-moteurs éventuels, des comportements inappropriés où leurs émotions prennent plus facilement qu'à l'acoutumée le pas sur le comportement rationnel ordinaire attendu dans ces cas-là.

Afin de pouvoir simuler les mécanismes à la fois cognitifs et affectifs à l'œuvre dans la genèse du comportement humain, il nous a donc été nécessaire de développer nos recherches selon plusieurs axes complémentaires. La modélisation comportementale est le premier sujet auquel nous nous sommes intéressé d'une part de part son importante utilisation dans les milieux professionnels et d'autre part parce que justement la société PerformanSe est éditrice de logiciels d'évaluation comportemental à destination des entreprises. La dimension émotionnelle moins souvent traitée dans les milieux professionnels car comme nous l'avons dit moins exprimer au travail a été notre deuxième axe d'étude avec les modèles là encore développée par les chercheurs pour expliquer l'origine de ce que l'on appelle les émotions. Enfin, nous sommes difficilement parvenu à collecter des travaux sur la dimension sociologique des émotions et sur la place de l'affect dans les rapports humains.

1.2 Les modèles comportementaux

1.2.1 Historique des modèles comportementaux

L'étude de l'individu et de son comportement est un champ de recherche ancien. Confondu au départ avec la philosophie, la psychologie est réellement née à la fin du XIX^{ème} siècle avec Freud (2005), père de la psychanalyse, Watson (1913), à l'origine du behaviorisme, Piaget (1950), opposé au behaviorisme et fondateur de la psychologie cognitive, et Hebb (2001), précurseur de la neuropsychologie. Les branches d'études se sont multipliées avec les progrès médicaux, biologiques, physiques et informatiques donnant accès à des données de plus en plus

²La majorité des individus atteints par ce traumatisme sont de hommes (à plus de 75%) ayant pour la plupart été victimes d'un accident de la route.

précises et variées pour expliquer le comportement. Parmi ces branches, la théorie des traits, sous domaine de la psychologie différentielle, cherche à déterminer la personnalité d'une personne et son comportement par rapport à un nombre fini de caractéristiques comportementales : les traits, dont la valorisation constitue un profil comportemental. Le modèle le plus répandu et celui sur lequel nous nous sommes en partie appuyé est celui du Big Five (Fiske, 1949; Tupes and Christal, 1961; Goldberg, 1990). Bien que cette prédiction du comportement et la notion même de personnalité aient été fortement remises en cause dans les années 70-80, de nouvelles méthodologies (Sinclair and Barrow, 1992) ont pu montrer la viabilité de ces théories et ont ainsi permis la résurgence des recherches sur le sujet et les outils liés au cours des deux dernières décennies.

1.2.2 Les outils d'évaluation comportementale

De nombreux outils d'évaluation de la personnalité sont utilisés en gestion des ressources humaines. Ils ont généralement pour but de construire le profil d'une personne afin d'aider le décideur. Un outil d'évaluation de la personnalité permet d'obtenir le profil d'une personne à partir de réponses à un questionnaire. Les intérêts de ce type d'outils sont multiples : aide au recrutement, aide à la mobilité interne, partie comportementale d'un bilan de compétences ... Il est entendu que ces outils ne doivent pas être utilisés à des fins discriminantes. En particulier, lors d'un recrutement, un tel questionnaire doit être une base d'entretien pour le responsable des ressources humaines et aucunement un outil de pré-sélection des candidats. La limite est parfois ténue et seul une formation et un accompagnement de l'évaluateur peut permettre d'éviter cet écueil. La plus grande difficulté réside dans le fait que chacun d'entre nous possède ses propres stéréotypes et qu'il est facile de rapidement les appliquer au profil comportemental de la personne évaluée pour la juger en bien ou en mal. C'est également avec cet objectif que la plupart des outils pérennes d'aujourd'hui ne se contentent pas de valoriser des paramètres dans un tableau, mais fournissent un descriptif nuancé des aptitudes et faiblesses de l'individu.

Les outils existants se divisent en questionnaires ouverts et questionnaires fermés. Pour les questionnaires ouverts, la personne est libre de ses réponses. Celles-ci sont ensuite analysées par un expert en psychologie qui va dessiner le profil comportemental de cette personne. Ces outils sont remis en cause car l'interprétation de l'expert est très subjective et variable, comme nous l'avons mentionné auparavant. C'est pourquoi ces questionnaires ne sont pas très nombreux. Nous citerons, par exemple, *Phrases* (Stein and Bransford, 1979) qui demande à l'utilisateur de compléter 50 phrases en 30 minutes, sous les yeux d'un examinateur. Ensuite, les réponses et le comportement de la personne lors de l'évaluation sont étudiées. Ce type d'outils a été très peu étudié car il est très difficile d'effectuer des analyses statistiques sur des questions ouvertes.

La deuxième catégorie d'outils est beaucoup plus répandue : il s'agit de questionnaires fermés. Ces questionnaires peuvent donc être informatisés facilement. Généralement, ils consistent en un ensemble de questions (les *items*) avec deux ou plus de choix possibles. Un ensemble de règles, du type de celles trouvées dans un système expert, ont été établies préalablement et permettent de construire un profil en fonction des réponses choisies. Un profil est généralement composé d'un nombre prédéterminé de traits de personnalités (les *dimensions*). De nombreux

outils basés sur ce fonctionnement existent, nous présentons les plus utilisés :

- *Sosie* ((Noci, 2003)) (d'ECPA) : 20 traits de personnalités évalués à partir de 98 groupes de 4 assertions,
- *PAPI* ((Kostick, 1960) Perception And Preference Inventory de Cubiks) :
 - le test classique : un choix pour 90 couples de phrases,
 - le test normatif : 126 assertions sur lesquelles l'utilisateur choisit entre « complètement en accord » ou « complètement en désaccord »,
- *MBTI* ((Myers, 1962; Myers and Myers, 1995)) (from Myers and Brigg) : 126 questions avec un choix entre deux réponses et un profil choisi parmi 16 profils prédéfinis,
- *PerformanSe Echo* (Philippé et al., 2004) : 70 questions avec deux réponses au choix et un profil selon 10 dimensions bipolaires déterminés grâce à un ensemble de règles,
- *Assess First* : 90 questions avec deux réponses au choix et un profil selon 20 dimensions comportementales et 5 familles.

Parmi tous ces outils, seuls *Sosie*, *MBTI* et *PerformanSe Echo* ont eu une réelle validation psychologique et statistique. Cependant, ils sont tous basés sur des théories psychologiques reconnues : la théorie de Jungian (Cowan, 1989; Myers, 1962), la théorie du « Big Five » (Goldberg, 1981; Wiggins, 1996) ou encore l'étude des motivations (George and Jones, 2002). Les études statistiques permettent d'apporter une validation complémentaire aux études comportementales sous-tendant les modèles psychologiques.

1.2.3 Le modèle comportemental PerformanSe

Présentation de la société PerformanSe

La société PerformanSe (PERfectionnement et FORMation au MANagement par Systèmes Experts) exerce depuis 1998 dans le domaine des outils d'évaluation comportementale. C'est avec la collaboration de professionnels en gestion des ressources humaines et d'universitaires spécialisés d'une part en extraction des connaissances et en gestion de la connaissance et d'autre part en psychologie différentielle qu'elle a conçu des outils informatisés utilisés pour la gestion individuelle et collective des ressources humaines dans le milieu professionnel. L'essentiel de son activité s'organise autour de l'évaluation et de la gestion des compétences comportementales grâce à une panoplie d'outils s'appuyant tous sur un socle commun : le modèle comportemental de PerformanSe articulé autour de 10 dimensions. PerformanSe a initialement racheté pour l'adapter l'outil américain Human Edge (aujourd'hui disparu) basé sur la théorie systémique du Big Five développée dans les années 50 par l'école de Palo Alto (PerformanSe, 2003). Ce modèle a ensuite été enrichi, structuré et affiné pour parvenir à une représentation cohérente de l'individu dans son milieu professionnel. L'ensemble des règles permettant d'établir le portrait comportemental d'un individu selon les dimensions de ce modèle a suivi la même évolution pour fournir un premier outil d'évaluation comportemental, décliné ensuite en un panel d'outils permettant d'accompagner l'individu au travail.

Au départ centrée sur le marché français, la société a depuis quelques années élargi ses horizons et adapté ces outils à d'autres cultures. Les différences entre les modèles culturels et

comportementaux des pays ont nécessité l'étalonnage de l'outil grâce à des études statistiques.

Conjointement à ses outils centrés sur l'individu, la société a également développé des outils pour déterminer *a priori* la dynamique d'un groupe en terme de forces d'attraction/répulsion. Néanmoins, aucun outil pour simuler l'évolution du comportement d'un individu, et plus encore, celle d'un groupe n'avait été créé. Intégrer le modèle comportemental de PerformanSe dans un agent est une des solutions pour laquelle nous avons opté pour parvenir à ce but.

La gamme des outils PerformanSe

PerformanSe propose aujourd'hui 5 outils adaptés spécifiquement aux différents besoins de la gestion des ressources humaines dans l'entreprise ;

- *Dialecho* : Dialecho est un outil d'évaluation des compétences comportementales basé sur deux approches complémentaires. L'auto-évaluation grâce au questionnaire Echo permet de dégager la façon dont la personne se perçoit. Le questionnaire Dialog permet lui l'évaluation par un tiers.
- *Oriente* : Cet outil permet à partir du profil établi par Dialecho de fournir le degré d'adéquation entre une personne et huit grands domaines d'activité : administrer, créer, échanger, produire, concevoir, gérer, argumenter et encadrer.
- *Aura Lead* : L'objet de cet outil est d'évaluer les capacités d'une personne à intégrer les attitudes de « Reconnaissance » dans ses pratiques managériales. L'objectif est de développer les attitudes managériales les plus adaptées et de dynamiser ainsi les facteurs individuels et collectifs de la performance.
- *Projequip* : Toujours en se basant sur l'expertise Dialecho, Projequip vise à comprendre la dynamique d'un groupe d'individus à priori afin de pouvoir constituer des équipes de travail efficaces.
- *Talents* : Cet outil est destiné à la gestion des compétences comportementales au sein d'une entreprise. Dans cette perspective, il peut ainsi favoriser la mobilité interne, diagnostiquer une adaptation à un environnement, fixer des axes de progrès, aider à la décision dans le choix des hommes, etc.

Le principal outil de PerformanSe est l'outil Dialecho (PerformanSe-Echo et PerformanSe-Dialog). Il est construit autour d'une base de connaissances et d'un système d'inférences permettant à un individu de répondre à des questionnaires et délivrant un bilan personnalisé sur les compétences comportementales de la personne (diriger, argumenter, ...). Ce sont les résultats obtenus par cet outil qui nous servent de jeux de données pour nos expérimentations, le but étant d'essayer de décrire des comportements remarquables.

L'outil PerformanSe Echo et le modèle en 10 dimensions

L'outil d'évaluation de la personnalité, Echo, développé par la société PerformanSe, est un questionnaire composé de 70 items. Chaque item a deux réponses possibles mais ce n'est pas forcément « oui » ou « non ». C'est un questionnaire *ipsatif*, c'est à dire à choix forcé. Par exemple, la figure 1.1 montre une question issue d'Echo et ses deux réponses possibles.

Question	Lorsque je suis en situation d'opposition
Réponses	Je discute fermement sans craindre les éventuelles tensions.
	Je recherche le consensus pour maintenir une atmosphère sereine.

FIG. 1.1 – Exemple de question ipsative du logiciel Echo.

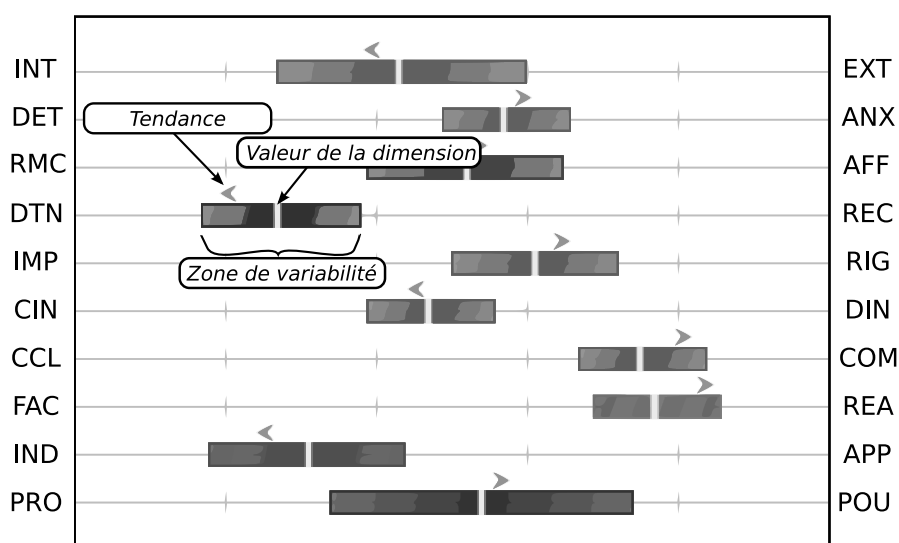


FIG. 1.2 – Profil bipolaire Echo sur 10 dimensions.

Une fois que toutes les questions ont été renseignées, l'outil propose le profil comportemental de l'évalué. Ce profil est construit à partir de 10 dimensions bipolaires détaillés dans le tableau 1.1. Chaque pôle d'une dimension est appelé un *trait* et a une valeur comprise entre 0 to 35. Chaque réponse à une question déclenche un ajout ou un retrait de points sur un ou plusieurs traits. Ensuite, à partir du score obtenu sur chaque couple de traits, un score de dimension bipolaire est calculé, compris entre 0 et 100. Afin de faciliter l'interprétation, ces scores sont discrétisés selon 3 zones : les petites valeurs (en dessous de 40) marquées -, les valeurs moyennes (entre 40 et 60) marquées 0 et enfin les valeurs élevées (au dessus de 60) marquées +. Par exemple, pour l'EXTraversion, les individus sont classés dans les 3 zones EXT-, EXT0 and EXT+. La figure 1.2 montre un exemple de profil comportemental bipolaire.

L'outil Echo est composé de 27000 règles qui permettent de construire, en plus du profil comportemental d'un individu, un rapport textuel à partir d'une base de textes de 2500 pages. Ce rapport considère les combinaisons entre les dimensions pour essayer de dresser les comportements de l'évalué. La figure 1.3 montre le fonctionnement du système. Ce modèle de per-

TAB. 1.1 – Les 10 dimensions comportementales.

Introversion (INT) <i>Exprime</i> : réserve, modestie, discrétion, risque de paraître froid, difficulté à communiquer capacité à se concentrer	Extraversion (EXT) <i>Exprime</i> : expansion, désir d'être remarqué, facilité d'expression, risque de mobiliser l'attention, tendance à l'intrusion
Détente (DET) <i>Exprime</i> : état de relaxation, sang-froid	Anxiété (ANX) <i>Exprime</i> : stress, inquiétude, pouvoir émotionnel, tension, concentration
Remise en Cause (RMC) <i>Exprime</i> : volonté de progression, sang-froid	Affirmation (AFF) <i>Exprime</i> : confiance en soi, convictions intimes, avis tranché
Détermination (DET) <i>Exprime</i> : distance aux autres, émotion, résistance passive	Réceptivité (REC) <i>Exprime</i> : ouverture sur les autres, goût pour l'écoute, empathie
Improvisation (IMP) <i>Exprime</i> : goût pour l'inattendu et l'adaptation, réactions spontanées aux événements, impulsivité	Rigueur (RIG) <i>Exprime</i> : environnement de travail structuré, sens de la méthode et planification, sens de la hiérarchie
Conformisme Intellectuel (CIN) <i>Exprime</i> : référence à des solutions bien connues, approche analytique, précision, difficulté à avoir une vue globale de situation, savoir expert	Dynamisme Intellectuel (DIN) <i>Exprime</i> : créativité, relations sociales, curiosité intellectuelle (nouvelles idées), compréhension globale des situations, risque de négliger des détails
Conciliation (CCL) <i>Exprime</i> : patience, recherche de relations sereines, esprit de consensus, capacité à agir en arbitre	Combativité (COM) <i>Exprime</i> : comportement réactif, recherche d'intérêts, esprit compétitif, attitude offensive, impatience
Motivation de facilitation (FAC) <i>Exprime</i> : plaisir immédiat <i>Principale crainte</i> : avoir trop de travail <i>Stimulus</i> : facilité, missions courtes <i>Satisfaction</i> : réussir des succès faciles <i>L'argent sanctionne</i> : la saisie des opportunités <i>Relation au temps</i> : projets à court terme <i>Voc.</i> : garder du temps, utiliser des raccourcis, privilégier le présent ...	Motivation de réalisation (REA) <i>Exprime</i> : persévérance et succès <i>Principale crainte</i> : être obligé d'abandonner <i>Stimulus</i> : projets difficiles <i>Satisfaction</i> : faire des efforts <i>L'argent sanctionne</i> : le mérite <i>Relation au temps</i> : projets à long terme, sentiment de culpabilité lié à la perte de temps <i>Voc.</i> : construire, persévérer, ténacité ...
Motivation d'indépendance (IND) <i>Exprime</i> : indépendance <i>Principale crainte</i> : être avalé par le groupe <i>Stimulus</i> : liberté personnelle <i>Satisfaction</i> : avoir chacun son territoire <i>L'argent sanctionne</i> : résultats individuels <i>Relation au temps</i> : se garder du temps pour soi <i>Voc.</i> : mesurer les conséquences de ses actes ...	Motivation d'appartenance (APP) <i>Exprime</i> : influence <i>Principale crainte</i> : être exclus <i>Stimulus</i> : la communauté <i>Satisfaction</i> : être en bons termes avec les gens <i>L'argent sanctionne</i> : résultats communs <i>Relation au temps</i> : donner son temps au groupe <i>Voc.</i> : consensus, solidarité ...
Motivation de protection (PRO) <i>Exprime</i> : le besoin de sécurité <i>Principale crainte</i> : ne pas avoir de garanties <i>Stimulus</i> : maintien des acquis <i>Satisfaction</i> : esprit en paix <i>L'argent sanctionne</i> : un droit acquis <i>Relation au temps</i> : être prévoyant <i>Voc.</i> : rester en terrain connu, éviter les surprises, s'organiser	Motivation de pouvoir (POU) <i>Exprime</i> : la prise de risque <i>Principale crainte</i> : ne pas avoir d'influence <i>Stimulus</i> : challenge, décider et mener <i>Satisfaction</i> : initiateur d'évènements <i>L'argent sanctionne</i> : le risque et la prise de responsabilité <i>Relation au temps</i> : envie de laisser une marque <i>Voc.</i> : être en position dominante, être ambitieux ...

sonnalité se fonde sur une théorie systémique développée dans les années 50 par l'école de Palo Alto (PerformanSe, 2003). Selon cette théorie, les traits de personnalité ont des liens avec les comportements et ils peuvent être représentés par un modèle en cinq dimensions (affirmation, bienveillance, émotivité, ouverture, sens des responsabilités) (PerformanSe, 2003). Elle définit la communication par des processus relationnels et interactionnels. Les rapports qui s'instaurent entre les individus priment sur les individus eux-mêmes. Il en découle que tout comportement est communication. L'ensemble des relations est considéré comme un système de communications qui interagissent. Il s'en dégage une logique de communication qui forme un ensemble de règles. Une communication perturbée entre un individu et les autres traduit un trouble des comportements.

Le modèle PerformanSe a adopté cette démarche en considérant l'individu comme un système en interaction avec un contexte (environnement, entourage, normes de travail, ...). L'individu est alors modélisé grâce à dix traits de personnalité en interaction entre eux et avec le contexte : sept dimensions comportementales (EXTraversion / INTroversion, ANXiété / DÉTente, COMbativité / ConCiLiation, RÉCeptivité / DÉTerMiNation, AFFirmation / ReMise en Cause, Dynamisme INtellectuel / Conformisme INtellectuel, RIGueur / IMProvisation) et trois motivations (RÉAlisation / FACilitation, APPartenance / INDépendance, POUvoir / PROtec-tion). C'est une extension du modèle du « Big Five » qui décrit une personnalité selon 5 dimensions que sont l'affirmation, la bienveillance, l'émotivité, l'ouverture et le sens des responsabilités (PerformanSe, 2003). Ce modèle est le résultat de plus de 40 années de travaux menés par une douzaine de chercheurs en psychologie tels que Catell, Fiske ou plus récemment Goldberg. Le modèle PerformanSe est un enrichissement de ce modèle avec en plus :

- l'étude des motivations qui permettent à un individu d'agir,
- l'approche systémique et comportementale qui considère un individu et son environnement comme un tout,
- la bipolarité : chaque dimension comporte deux pôles (traits de personnalité opposés) vers lesquels l'individu peut plus ou moins tendre selon le contexte et sa plage de variabilité.

1.2.4 Spécificités des personnes cérébrolésées

Dans le cadre initial de notre étude, la simulation concernait le comportement de personne cérébrolésée sur des chaînes de production. Employés par dans un atelier d'aide à la réinsertion dans le milieu professionnel des traumatisés crâniens, l'objectif était de simuler les comportements d'équipe afin de fournir une aide à la décision pour la détermination des équipes offrant le meilleur compromis entre efficacité au travail et progrès psychologique des personnes. Afin de parvenir à ce but, il est nécessaire de prendre en compte les spécificités des personnes atteintes de ce type de traumatisme. La caractéristique la plus communément partagée entre les personnes cérébrolésées est que les filtres émotionnels disparaissent ou deviennent moins performants (Der Linden et al., 1999). De plus, les personnes traumatisées souffrent de troubles de la mémoire et de l'attention. Les impacts du traumatisme montrent l'importance des émotions sur le comportement. En effet, de récentes découvertes en neuropsychologie (Damasio, 1994) ont mis en évidence la nécessaire influence des émotions sans lesquelles il est impossible de réfléchir et de décider. Cela explique également pourquoi nous avons pris en compte les émotions

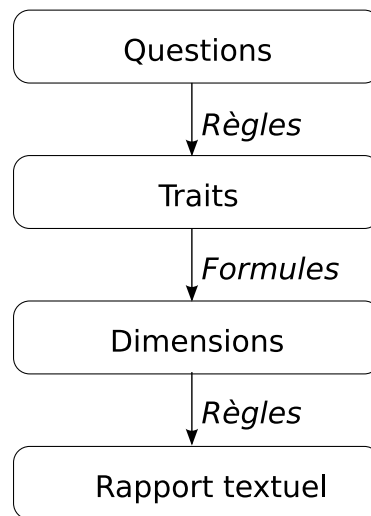


FIG. 1.3 – Système Echo à base de règles.

dans notre modèle d'agent.

En outre, le contexte est bien adapté à la modélisation et à la simulation du comportement humain. Les traumatismes crâniens entraînent des réactions disproportionnées et des comportements inadaptés aux situations. Ainsi, la première étape de la simulation est de parvenir à la restitution de ces réactions spontanées et imprévisibles, qui sont plus fréquentes chez les personnes souffrant de ce handicap puisque leurs filtres émotionnels fonctionnent mal.

1.3 Le système cognitif humain

La cognition regroupe les divers processus mentaux allant de l'analyse perceptive de l'environnement à la commande motrice (en passant par la mémorisation, le raisonnement, les émotions, le langage, ...). Cette définition dépasse donc le seul cadre de la cognition humaine ou animale mais comprend aussi les processus à l'œuvre dans des systèmes artificiels, comme les ordinateurs. À la suite de la révolution cognitiviste, la perspective dominante depuis le milieu du XX^{ème} siècle regroupe sous le terme de cognition les fonctions dont est doté l'esprit humain et par lesquelles nous construisons une représentation opératoire de la réalité à partir de nos perceptions, susceptible en particulier de nourrir nos raisonnements et guider nos actions. En d'autres termes, la cognition regroupe l'ensemble des fonctions qui permettent à un être d'appréhender son environnement et d'y évoluer de manière sensée (perception, langage, mémoire, raisonnement, décision, mouvement ...). Pour le cas spécifique de l'être humain, on parle également de fonctions cognitives supérieures pour désigner les facultés que l'on retrouve chez l'homme mais pas - avec les mesures de précautions qui s'impose autour de cette affirmation - chez l'animal comme le raisonnement logique, le jugement moral ou esthétique. D'autres courants de recherche, plus proche de la systémique, caractérisent au contraire la cognition comme

un phénomène essentiellement dynamique et émergent, produit global d'une activité cérébrale multiple. Dans notre cas, nous nous sommes attachés à la première vision cognitiviste mieux adaptée à une modélisation structurée d'une entité informatique.

Les récentes études englobent également sous le terme cognition des processus mentaux comme l'affect, l'instinct ou l'éthique jusque là laissés en marge car considérés comme ne faisant pas partie ou perturbant le système cognitif. Il s'est pourtant, avéré grâce aux études menées durant les 40 dernières années, qu'il était impossible de dissocier ces aspects du comportement humain. Ainsi, en économie expérimentale, il a été montré que les êtres humains peuvent agir irrationnellement dans des situations où pourtant la raison « à froid » ordonnerait d'agir autrement. Ou encore, en neuropsychologie, les patients souffrant d'une lésion cérébrale dans le cortex préfrontal sont incapables de réagir correctement à une situation émotionnelle tout en étant parfaitement capable de raisonner dans l'abstrait. Nous illustrerons d'ailleurs les parties suivantes avec les effets de traumatismes sur les fonctions mentales présentées. Le terme cognition inclut donc aujourd'hui un ensemble très vaste de processus mentaux. Dans les parties qui vont suivre, nous avons néanmoins conservé la distinction entre ce que l'on pourrait appeler le cœur et la raison, ceci afin de mieux illustrer l'apport de la prise en compte de la dimensions affective dans la modélisation d'une entité virtuelle. Nous présenterons donc dans un premier temps le système cognitif « raisonné », puis dans un second temps le système affectif et son rôle dans le processus cognitif. Nous allons dès à présent détailler les cinq principales fonctions cognitives.

1.3.1 L'attention

L'attention est une fonction cognitive complexe. Elle correspond à un processus de sélection d'un événement extérieur (son, image, odeur, ...) ou intérieur (pensée) et du maintien de ce dernier à un certain niveau de conscience. On distingue trois grands types d'attention (Rozec, 2003) :

- La vigilance permet le maintien de l'attention dans les tâches de longue durée aux stimuli peu fréquents et attention soutenue est son pendant pour les tâches de longue durée aux stimuli fréquents et réguliers.
- L'attention sélective est la capacité à sélectionner, dans l'environnement, un stimulus particulier et de porter sur lui les ressources attentionnelles disponibles, en prenant soin que les autres stimuli de l'environnement ne viennent pas perturber la tâche en cours. Elle permet donc de se couper mentalement de l'environnement sans devoir s'isoler physiquement. Pour cela, un modèle a été proposé par Tipper (1991) où deux mécanismes sont mis en œuvre. Tout d'abord, l'information pertinente est sélectionnée puis les stimuli non pertinents pour celle-ci sont inhibés. Il est à noter que l'attention n'est pas stable. Elle ne peut rester soutenue tout au long d'une activité, elle connaît des états de relâchement.
- L'attention partagée est la capacité à répartir ses ressources attentionnelles sur plusieurs tâches. L'âge, la fatigue ou encore le stress rendent plus sensible aux interférences et donc moins performant en situation de double tâche.

Une lésion du système attentionnel entraîne une diminution du temps de réaction, une lenteur dans le traitement de l'information pour les tâches demandant une réponse rapide, des difficultés d'utilisation des stimuli visuels, de balayage visuel et de poursuite oculaire. On constate chez les personnes traumatisées crâniennes une vigilance diminuée, des difficultés d'attention soutenue et des troubles d'attention sélective et partagée lors d'une demande cognitive importante et complexe (Rozec, 2003). L'affectation des capacités attentionnelles est importante à prendre en compte car l'attention est le pré-requis au bon fonctionnement de la mémoire, du langage, de la perception et de toutes les activités demandant de la concentration.

1.3.2 La mémoire

La mémoire est sans doute la fonction cognitive la plus largement sollicitée dans la plupart de nos actes. Elle est en permanence mise à contribution de manière volontaire ou non. Elle représente la connaissance au sens large (souvenirs, connaissances culturelles, procédures motrices, ...) de chacun. Il existe différentes mémoires selon le type de connaissance mémorisée que l'on souhaite atteindre.

Tout d'abord, la mémoire sensorielle est la plus éphémère. Elle capte toutes les nouvelles informations reçues pendant quelques centaines de millisecondes seulement.

Puis la mémoire à court terme ou mémoire de travail prend le relais. C'est la capacité à maintenir de l'information dans la mémoire immédiate et à la manipuler en fonction d'un but ainsi que la capacité à effectuer plusieurs tâches cognitives simultanément (Rozec, 2003). Elle conserve l'information environ une minute et permet d'enregistrer en moyenne sept éléments à la fois³. Elle intervient, par exemple, quand on lit. Elle nous permet de retenir la phrase lue afin d'assurer une cohérence avec la suite de la lecture. Selon Baddeley and Hitch (1974), la mémoire de travail est composée de trois entités : l'administrateur central de capacité limitée (environ 7 mnèmes), une unité de stockage de l'information verbale et une unité de stockage de l'information visio-spatiale. Les conséquences possibles d'un traumatisme sont donc un oubli de ce que l'on vient de dire ou entendre, de l'information visuelle sur un parcours, ...

Enfin, on sollicite la mémoire à long terme lorsque l'on souhaite retenir une information plus longtemps ou accéder à une information plus ancienne. Sa contenance et sa durée de conservation sont illimitées. Dans la mémoire à long terme, on distingue les mémoires épisodique (événements situés dans un contexte spatio-temporel comme une réunion), sémantique (connaissances générales du type capitales géographiques) et procédurale (apprentissage de procédures motrices ou cognitives pour automatiser une tâche) (Tulving, 1972).

1.3.3 Le langage

Le langage constitue la caractéristique humaine par excellence. Il consiste à l'assemblage d'un certain nombre de phonèmes afin de communiquer et de transcrire (à l'écrit (lecture et écriture) et à l'oral (compréhension et expression)) des idées ou des sentiments (fonction du

³On parle de *mnèmes*.

langage également partagée par les animaux). Le langage repose d'une part sur la capacité physique à produire des sons intelligibles constituant les bases de la communication orale, ou à écrire quant il s'agit de communication écrite, et d'autre sur la capacité mentale à mémoriser et associer de manière logique les unités promordiales du langage. Le langage est fortement lié à l'ensemble des autres fonctions cérébrales. Ainsi, il est fréquent que l'altération de l'une d'entre elles (dans le cas d'un traumatisme crânien notamment) ait des répercussions sur lui : perte de mémoire occasionnant une perte du langage totale ou partielle, trouble du contrôle nerveux entraînant une invalidité totale ou partielle dans le mécanisme d'articulation linguistique, etc.. Indépendamment de ceux ci, une lésion au niveau du centre de contrôle du langage peut entraîner des troubles plus spécifiques troubles du langage qui peuvent se situer à différents niveaux, être afférents (perception et compréhension) et/ou efférents (expression) : persévérations verbales, stéréotypies verbales, troubles d'évocation lexicale, paraphasies. On note également assez souvent à une dissonance comportementale visible au niveau du langage par une inadéquation entre le ton (de calme à agressif) et le niveau de langage (soutenu, familier, vulgaire) employé et le contexte.

Une question philosophique toujours ouverte est de savoir si la pensée préexiste au langage ou si son existence est conditionnée par sa formalisation par le langage. La modélisation d'une entité virtuelle, et l'informatique en générale, sont directement concernée par cette question puisque toute conception informatique s'appuie sur des langages de développement, de déclaration, de modélisation formels. Tout comportement « intelligent » d'une entité virtuelle est donc totalement conditionné par sa formalisation par un langage. Dans le monde informatique, le langage préexiste donc à tout le reste.

Dans le contexte des systèmes multi-agents, le langage, ou plutôt la communication, est une notion centrale. C'est sur elle que s'appuie l'ensemble des interactions entre les entités, et donc la dynamique du système (*c.f.* 2.3.4). Bien entendu, exceptés les rares cas où on fait usage de la reconnaissance vocale pour interpréter les flux échangés, cette communication s'effectue dans une forme textuelle normalisée. Néanmoins, cette communication correspond la plupart du temps à des messages oraux virtuels échangés par les entités. Certaines notions psychologiques afférentes au langage peuvent donc être appliquées à la modélisation des agents. La communication entre agents de types requête-réponse formatées reste toutefois limitée en expressivité et il inenvisageable à l'heure actuelle de doter une entité informatique de capacités d'expression et de compréhension satisfaisantes.

1.3.4 La vision et l'imagerie mentale

La vision est une fonction extrêmement complexe si l'on pense au nombre d'informations à traiter pour analyser une scène visuelle. Elle nous permet d'identifier les objets de l'environnement et leurs positions relatives. Elle assure donc un rôle majeur pour s'orienter et interagir avec l'environnement. L'aspect cognitif de la vision regroupe à la fois la perception courante de l'environnement et l'imagerie mentale. On parle d'imagerie mentale visuelle pour l'activité cognitive qui permet de se représenter quelque chose en son absence. Cette représentation interne peut être volontaire ou suscitée (son, odeur, ...).

La vision a également un rôle central dans le processus d'interaction sociale. Un ensemble important de stimuli conscients ou inconscients sont en effet émis par les individus et perçus grâce à la vision. L'expression faciale, mais également l'attitude globale d'une personne, perçues grâce à la vision, sont interprétées volontairement ou non par le cerveau et contribuent ainsi aux interactions sociales. Cet aspect de la vision est crucial dans le contexte de la modélisation informatique d'un agent émotionnel. En effet, si une partie des stimuli reçus sont analysés de manière cartésienne, la plupart ont un impact sur l'affect.

L'imagerie mentale visuelle est une activité cognitive qui permet de percevoir quelque chose en son absence. Elle n'est donc pas exclusivement visuelle (son, odeur, ...). Là encore, dans notre cas, plus de détails ne s'imposent pas.

1.3.5 Les fonctions exécutives

Les fonctions exécutives, selon Rabbitt (1997) jouent sept rôles qui sont :

- gérer la nouveauté, lorsqu'une tâche est nouvelle et nécessite la formulation d'un but,
- rechercher de manière délibérée de l'information en mémoire,
- initier de nouvelles séquences de comportement tout en interrompant des séquences en cours ou en inhibant des réponses habituelles,
- empêcher la production de réponses non appropriées au contexte,
- coordonner la réalisation simultanée de deux tâches,
- détecter et corriger des erreurs,
- maintenir l'attention de façon soutenue sur de longues périodes de temps.

Sur le plan physiologique, on associe les fonctions exécutives au système frontal du cerveau humain. Elles sont impliquées dans l'attention soutenue, sélective, alternée et partagée, dans la mémoire de travail, des séquences temporelles, du contexte et prospective. Elles interviennent aussi dans la régulation de l'humeur et des émotions, la capacité de planification. La coordination de l'ensemble des fonctions exécutives est assurée par le Système Attentionnel Supérieur (SAS). Il assure, à partir de l'ensemble des fonctions du système cognitif précédemment décrit, production d'un comportement de l'individu adapté à la situation. Son dysfonctionnement peut donc se traduire par des réactions inadéquates du sujet aux situations nouvelles, une modification de la tonalité émotionnelle (pseudo-dépression, pseudo-psychose), une altération de la perception et l'expression des émotions ou encore une modification de la personnalité. Les troubles des fonctions exécutives sont donc souvent la cause de difficultés d'adaptation en milieu familial, social ou professionnel (Rozec, 2003).

Le modèle de Norman and Shallice (1980) (figure 1.4) tente de décrire le fonctionnement du système exécutif. L'unité principale du modèle est le schéma d'action (structure de connaissances qui contrôle les séquences d'actions). Lorsque deux actions entrent en conflit, le gestionnaire de conflits sélectionne la tâche la plus en adéquation avec les objectifs. Le gestionnaire de conflits a donc pour fonction de choisir le schéma d'actions le plus pertinent. Le SPC est un processus automatique qui est impliqué lors de l'activation de schémas cognitifs propres aux activités familières ou routinières. Le SAS, par contre, est activé pour trouver une solution à une situation inhabituelle. Il n'agit pas directement sur l'action mais plutôt sur le gestionnaire

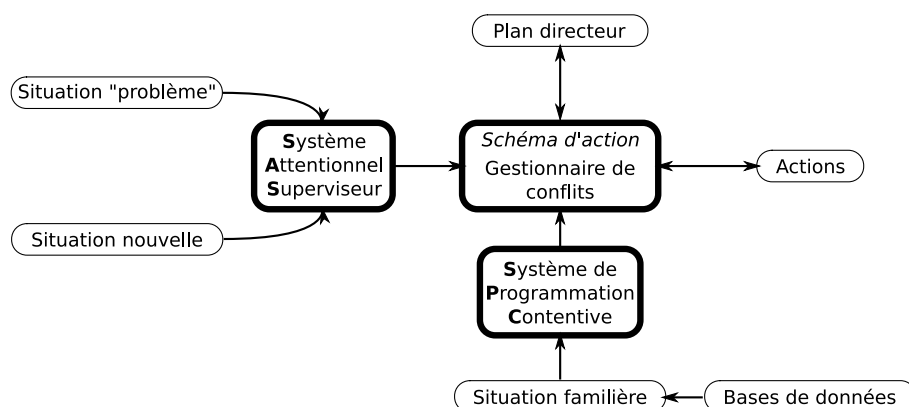


FIG. 1.4 – Modèle de contrôle attentionnel de Norman and Shallice (1980).

de conflits en ajoutant l'activation ou l'inhibition nécessaire à rendre un schéma prioritaire. Les situations qui activent le SAS sont les situations impliquant un processus de planification ou de prise de décision, les situations impliquant la correction d'erreur, les situations dans lesquelles les réponses ne sont pas bien apprises ou contiennent de nouvelles séquences d'actions, les situations difficiles techniquement ou dangereuses et enfin, les situations qui exigent de contre-carrer une réponse apprise ou de résister à la tension (Rozec, 2003).

1.4 Les émotions

1.4.1 Définition

« Tout le monde sait ce qu'est une émotion, jusqu'à ce que vous lui demandiez de la définir. »⁴

(Fehr and Russell, 1984)

En effet, l'étude des émotions appartient à deux traditions différentes, la philosophie et la biologie. Le philosophe s'intéresse à la nature même des émotions, à leurs relations avec les autres ressorts de la vie affective que sont les passions et les sentiments et à ce qui les diffère de la vie rationnelle et réfléchie. Pour le biologiste, par contre, l'émotion n'est accessible qu'au travers des modifications objectives qui l'accompagnent (Dantzer, 2003). Si on ajoute à cela le fait que les émotions puissent être envisagées selon plusieurs théories, que nous verrons par la suite, il en résulte une multitude de définitions proposées. Nous allons nous attacher à une définition assez générale que nous préciserons en faisant la distinction entre émotion et sentiment.

Mandler (1992) définit l'émotion comme étant un événement conscient créé par la combinaison d'évaluations cognitives (bon, mauvais, plaisant, déplaisant, nocif, désirable, ...) et de l'activation physiologique. Ensemble, ces composantes constituent l'expérience de l'émotion.

⁴ "Everyone knows what an emotion is, until asked to give a definition."

Mandler s'inscrit dans une approche cognitivo-physiologique de l'émotion que nous verrons par la suite.

Les émotions sont caractérisées par des sensations plus ou moins nettes de plaisir ou de déplaisir. Elles ne sont pas seulement cérébrales car elles sont aussi accompagnées de modifications physiques et somatiques. C'est d'ailleurs cette caractéristique qui les distingue des simples sentiments. On distingue trois grands types d'émotion (Dantzer, 2003) :

- les émotions fondamentales sont une réaction à un événement extérieur réel ou imaginaire (le dégoût et la peur, par exemple),
- les émotions dérivées sont fondées sur l'émotion engendrée par l'image que l'on a de la conscience de l'autre (le mépris est le dégoût pour les prétentions d'un autre, la méfiance est la peur du mystère),
- les émotions tierces naissent de la conscience de soi face au regard de l'autre (la honte est le mépris de soi, la timidité est la peur de sa propre valeur telle qu'elle est perçue par les autres).

1.4.2 Théories des émotions

L'approche physiologique

Les théories physiologiques mettent l'accent sur le type de relations possibles entre l'émotion, en tant qu'état mental ou cognitif, et son expression somatique (Dantzer, 2003). Nous allons présenter les trois grandes théories physiologiques, celles de James (1884), de Lange (1885) et de Cannon (1927) (recensées par Christophe (1998)).

Les théories de James et de Lange, finalisées respectivement en 1884 et 1885, consistent à opérer une assimilation des états émotionnels et des perceptions corporelles. Ils proposent des idées similaires. Premièrement, le processus émotionnel suit la séquence suivante : stimulus, réponses physiologiques, sensation de ces changements, émotion. Pour illustrer son propos, James donne l'exemple de la rencontre avec un ours (stimulus). L'Homme commence à courir (changements physiologiques) puis il perçoit ces changements inhérents à la course et alors il a un sentiment de peur. Ensuite, ils pensent tous les deux que les réponses corporelles dans les émotions sont constituées de changements physiologiques différents selon les émotions. Le postulat central est que la perception des changements périphériques constitue en soi l'émotion, chaque type d'émotion correspondant à une structure différente de sensations. Parmi, les activations physiologiques, on trouve les changements dans l'expression faciale qui permettent à autrui de reconnaître l'émotion qui nous submerge (*c.f.* figure 1.5). Cependant, ces deux théories ont un point de désaccord important sur l'existence ou non d'un centre émotionnel spécifique. Selon James, l'émotion est déterminée par les centres moteurs et sensoriels alors que Lange pose l'existence d'un centre spécifique du cerveau, le centre « vasomoteur » qui commande le système vasculaire. Les changements vasculaires seraient donc les premiers processus physiologiques des émotions.

La théorie de Cannon (1927) est, elle, centraliste. Il affirme que les émotions sont induites par l'excitation du thalamus et que les changements vasculaires n'interviennent pas dans l'in-

duction de l'émotion. Pour cela, il développe quatre arguments. Tout d'abord, l'altération des connections entre les viscères et le système nerveux ne modifie pas le comportement émotionnel. Puis, les viscères sont relativement insensibles. Ensuite, les mêmes changements apparaissent dans des états non-émotionnels. Enfin, l'induction artificielle de changements typiques ne produit pas les émotions attendues. Pour Cannon, l'activation émotionnelle est largement indifférenciée entre les états émotionnels et non-émotionnels.

Ces trois théories mettent en évidence le fait que toute émotion est incontestablement accompagnée de manifestations physiologiques. Cependant, les aspects sociaux ne sont pas du tout abordés. L'environnement n'est pas pris en compte dans l'intensité des manifestations physiologiques ressenties⁵. De plus, l'évaluation subjective des stimuli n'est pas prise en compte. C'est en réponse à ces lacunes que se sont développées les théories cognitives.

Les théories néo-darwiniennes

Les théories évolutives spéculent sur les éléments qui confèrent un avantage sélectif aux capacités d'expression des émotions. La principale contribution a été apportée par Darwin (1872), selon qui les émotions seraient une sorte de comportement rudimentaire, résiduel d'un système plus complexe qui se serait modifié au fil de l'évolution. Les émotions auraient donc une qualité primitive adaptative liée tant au passé de notre espèce en termes d'évolution qu'à notre histoire personnelle. Trois principes explicitent sa démarche :

le principe d'utilité ou habitudes associées : les expressions émotionnelles sont à l'origine des actes utilitaires qui rempliraient une fonction adaptative par rapport à l'environnement,

le principe d'opposition ou antithèse : les émotions antagonistes engendrent des comportements opposés,

le principe d'action directe de l'excitation nerveuse sur le corps : les circuits nerveux contrôlant l'expression motrice des émotions sont en quelque sorte précâblés et ils deviennent de plus en plus perméables au fur et à mesure qu'ils sont sollicités.

Darwin a effectué la première classification des émotions déterminée par des configurations expressives faciales et corporelles spécifiques. Les théories néo-darwiniennes se sont essentiellement focalisées sur la détermination des émotions de bases ou fondamentales en étudiant les expressions faciales émotionnelles. Les quatre principaux auteurs sont Tomkins (1962, 1980, 1986), Izard (1971, 1977, 1990, 1997), Plutchik and Kellerman (1980) et Ekman et al. (1972); Ekman and Friesen (1982); Ekman (1982) (tableau 1.2). Ekman et al. (1972) a déterminé six émotions ayant une expression universelle (*c.f.* figure 1.5). Ekman a ensuite étendu ces 6 émotions à 15 en 1999, en introduisant principalement des déclinaisons des 6 émotions originelles (par exemple, la rage qui est une extension de la colère). Ces diverses conceptions mettent l'accent sur la relation entre une configuration expressive faciale et une émotion spécifique, l'expression faciale ayant un rôle déterminant dans le déclenchement de l'expérience subjective émotionnelle. Comme dans les théories cognitives, ces expressions émotionnelles auraient une

⁵On peut penser aux mouvements de foule, par exemple.

fonction adaptative. De plus, elles permettraient de communiquer à autrui son état émotionnel interne. Cependant, il est à noter qu'un individu peut cacher ou fausser son état émotionnel, l'expression faciale n'est donc qu'un indicateur infallible de l'état émotionnel (Dantzer, 2003).

TAB. 1.2 – Liste des émotions fondamentales proposées par chaque auteur (Christophe, 1998).

	Tomkins	Izard	Plutchik	Ekman
Colère	✓	✓	✓	✓
Dégoût	✓	✓	✓	✓
Joie	✓	✓	✓	✓
Peur	✓	✓	✓	✓
Surprise	✓	✓	✓	✓
Tristesse		✓	✓	✓
Mépris	✓	✓		✓
Honte	✓	✓		✓
Intérêt	✓	✓		✓
Culpabilité		✓		✓
Acceptation			✓	
Espérance			✓	
Anxiété	✓			
Embarras				✓
Respect				✓
Excitation				✓

L'approche cognitive

Les théories cognitives de l'émotion ont longtemps débattu de la relation entre émotions et fonctions cognitives. Les développements les plus récents insistent sur le rôle des facteurs de personnalités et de l'expérience antérieure (Dantzer, 2003). De manière générale, on peut définir la cognition comme l'ensemble des connaissances que le sujet a du monde et de lui-même et comme l'ensemble des processus qui permettent d'acquérir ces connaissances. Depuis les années 60, les études se sont tournées vers le rôle des processus cognitifs dans l'élaboration de l'émotion.

Schachter and Singer (1962) furent parmi les premiers à étudier le processus émotionnel comme résultant du jeu de deux grands facteurs : l'activation physiologique et l'activité cognitive. La situation est interprétée en fonction des expériences passées du sujet. Celles-ci fournissent la matière à partir de laquelle on peut comprendre et nommer ses sentiments. Si le sujet possède une cognition explicative de son état d'activation, il identifiera immédiatement son état émotionnel. Sinon, il cherchera à interpréter son état interne en se comparant à autrui⁶ (c.f. figure 1.6).

⁶Ils introduisent la notion de besoin évaluatif en s'appuyant sur la théorie de la comparaison sociale.

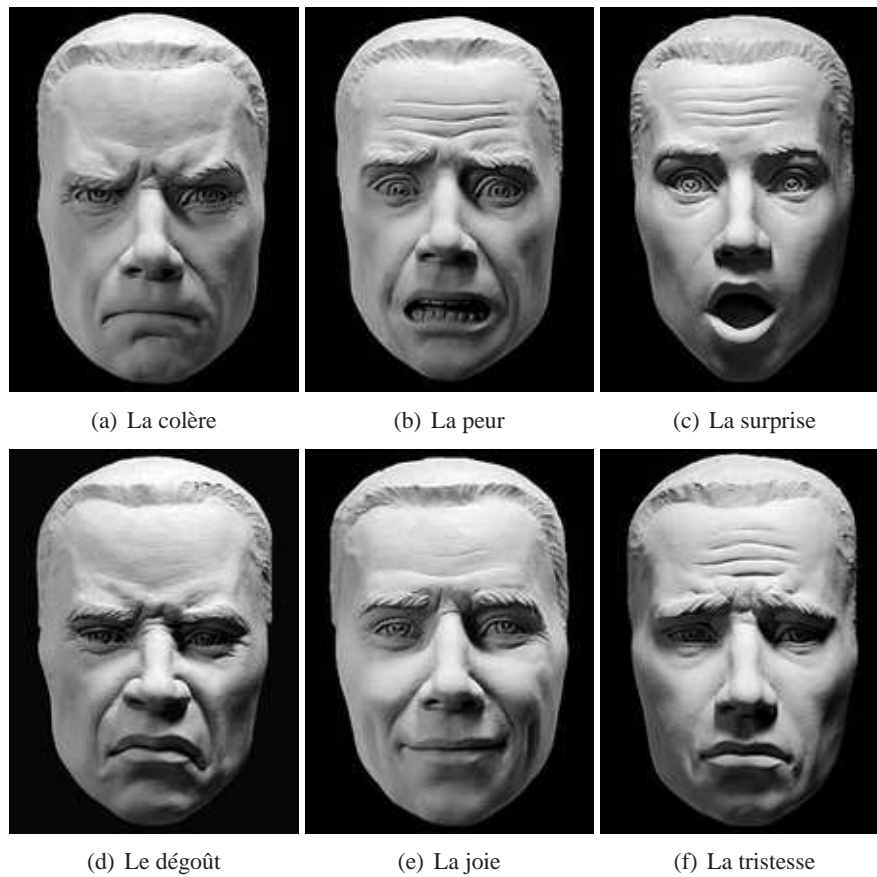


FIG. 1.5 – Expressions universelles d'Ekman.

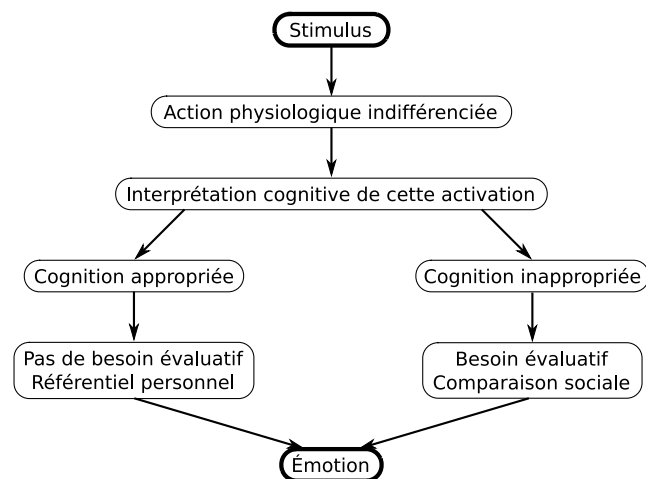


FIG. 1.6 – Théorie des émotions de Schachter and Singer (1962).

Une des théories fondamentale de l'approche cognitiviste est la théorie de l'*appraisal*⁷ (Frijda, 1986; Smith and Lazarus, 1990)). Les théories de l'évaluation envisagent la cognition de la séquence émotionnelle comme la perception et l'évaluation de la signification d'un événement pour le bien-être d'une personne. Arnold (1950) en est le précurseur. Il fait appel au concept de mémoire des expériences émotionnelles antérieures et à l'évaluation des conséquences possibles de la situation (*c.f.* figure 1.7). L'environnement social n'est pas directement impliqué. Il peut, tout au plus, être considéré comme stimulus de la séquence émotionnelle. Cependant, les expériences sont liées au contexte social de l'individu.

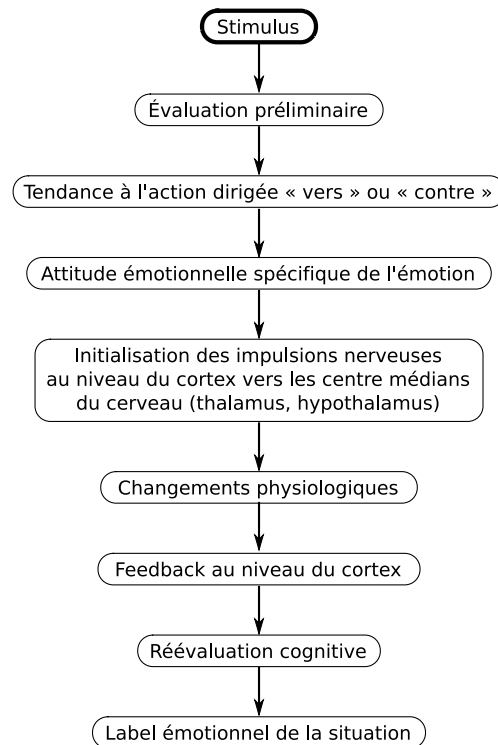


FIG. 1.7 – Théorie des émotions de Arnold (1950).

Lazarus (1968) considère que les émotions découlent de l'influence mutuelle d'un sujet et de son environnement et une telle interaction engendre des coûts (émotion négative) ou des bénéfices (émotion positive). Cette évaluation serait influencée par les normes, les règles et les caractéristiques sociales de l'environnement. C'est sur cette théorie de l'*appraisal* que des modèles émotionnels, tel que le modèle OCC (*c.f.* chapitre 1.4.4), aujourd'hui appliqués à l'informatique ont été construits.

La principale limite de l'approche cognitive est que les chercheurs se sont surtout focalisés sur l'évaluation cognitive en limitant l'émotion à un processus individuel même si les aspects sociaux intervenant dans la séquence émotionnelle n'ont pas été totalement négligés. Les modèles cognitifs ne placent pas l'individu et l'environnement au même plan. Ce dernier n'est

⁷Théorie de l'évaluation, en français.

vu que comme une source potentielle d'information. L'émotion est une construction purement interne. Il apparaît pourtant que le processus d'évaluation du stimulus dépend de l'histoire du sujet (expériences passées, normes ...). La socialisation, la culture des individus sont donc des facteurs à prendre en compte dans la séquence émotionnelle. À l'opposé de ces modèles, l'approche sociale constructiviste (Berger and Luckmann, 1966; Harre, 1989) considère à l'inverse, et de manière également insatisfaisante, que l'émotion est le seul produit du contexte social. L'approche cognitive reste néanmoins la plus utilisée dans le domaine de l'informatique du fait d'une implémentation relativement simple de cette théorie. Le principe est de lier l'émergence d'une émotion à la façon dont la personne perçoit son environnement. Pour dessiner un parallèle rapide avec le modèle BDI, les émotions émergeraient après que les connaissances de l'agent aient été mises à jour.

1.4.3 Le rôle des émotions

Dans la littérature, on attribue trois grands rôles aux émotions. Elles sont, à la fois, un processus adaptatif, ont un rôle organisateur et une valeur de signal. C'est ce que nous allons détailler maintenant.

Des processus adaptatifs

Plusieurs auteurs, comme Lazarus (1968) ou Scherer (1981), se sont également intéressés aux fonctions de l'émotion. Celles-ci sont essentiellement perçues comme un processus adaptatif qui permet à l'individu de faire face à la situation émotionnelle. Comme dans les théories des émotions, là encore, même si les auteurs envisagent les interactions du sujet avec son environnement, la dimension sociale est écartée. Tout se passe comme si les émotions ne remplissaient que des fonctions individuelles et ne servaient pas au groupe dans son ensemble. Toutefois, le concept d'utilité sociale a été abordé dans la lignée des théories évolutionnistes. L'universalité des émotions à travers les cultures chez l'Homme et dans le règne animal laisse penser que les émotions sont des processus adaptatifs favorisant la survie de l'individu et de l'espèce. Les petites peurs de l'enfant (peur du noir, peur d'être seul, ...), par exemple, sont le résultat d'une longue série d'associations dans l'histoire de l'humanité entre ces situations et un danger potentiel (Dantzer, 2003).

Un rôle organisateur

Ensuite, les émotions jouent un rôle organisateur important dans l'évaluation du monde. En effet, elles nous permettent de catégoriser le monde extérieur en pratiquant des dichotomies comme agréable-désagréable ou encore bénéfique-nocif. Cette catégorisation nous aide à choisir et oriente donc notre action.

Une valeur de signal

Enfin, la dernière grande fonction des émotions est représentée par la valeur de signal associée à chacune d'elles. En effet, en utilisant la posture, l'expression faciale, la gestuelle et l'expression vocale de son entourage, on peut accéder à l'état émotionnel des personnes. Ainsi, un individu peut prévoir son action, réagir de façon appropriée à une situation en se basant sur l'état émotif d'un autre individu. Le fait que l'on puisse feindre une émotion pour tromper délibérément son entourage est certainement la meilleure illustration de cette fonction communicative de signal qu'a l'émotion.

1.4.4 Le modèle OCC

Le modèle OCC a été mis au point par Ortony et al. (1988). Leur théorie se base sur la spécification de la structure psychologique des émotions en les liant à des situations personnelles ou non. Ils ont construit une théorie cognitive concernant les origines des émotions. Leur approche est particulièrement intéressante car elle permet de structurer les émotions, domaine aux influences très diverses (biologie, psychologie, cognition, ...) et donc assez confus. De plus, elle vise à l'obtention d'un modèle des émotions orienté implémentation informatique. « ..., nous aimerions établir les bases d'un modèle émotionnel réductible en temps polynomial. En d'autres mots, nous souhaiterions un ensemble d'émotions qui pourrait être utilisé dans un système d'Intelligence Artificielle (IA) qui pourrait, par exemple, être capable de raisonner sur les émotions. »⁸ C'est pour ces deux principales raisons pour lesquelles ce modèle reste aujourd'hui le plus utilisé dans l'Intelligence Artificielle.

Leur analyse découpe les émotions en trois grandes classes que sont les émotions en réaction à un événement, les émotions en réaction à un individu et celles en réaction à un objet. À chaque émotion sont associées les conditions d'activation de celle-ci et les variables qui vont influencer sur l'intensité de l'émotion. Par exemple, la joie est déclenchée par l'arrivée d'un événement attendu et souhaité et l'intensité de cette joie est fonction de la désirabilité de l'événement déclencheur. Les conditions de déclenchement de chacune des émotions sont présentées en annexe 1.4.4. Leur modèle propose 11 couples d'émotions classés dans six catégories (émotions de bien-être, liées aux ressentis pour les autres, liées à un événement éventuel, d'attribution, composées et d'interaction)⁹, comme le montre la figure 1.8.

Le modèle OCC a d'autre part été éprouvé dans plusieurs projets : Oz, MRE, ... Il est bien adapté à l'informatique puisque les émotions sont déclenchées par un stimulus, ce qui fournit déjà un ensemble de règles de génération des émotions. De plus, il comporte une base de 22 émotions (11 couples) mais la structure est facilement extensible ou à l'inverse simplifiable étant donné que les émotions sont catégorisées vis-à-vis de l'environnement. Leur nombre peut

⁸“..., we would like to lay the foundation for a computationally tractable model of emotion. In other words, we would like an account of emotion that could in principle be used in an Artificial Intelligence (AI) system that would, for example, be able to reason about emotion.”

⁹*well-being, fortunes-of-others, prospect-based, attribution, attraction and well-being/attribution compounds emotions*, en anglais.

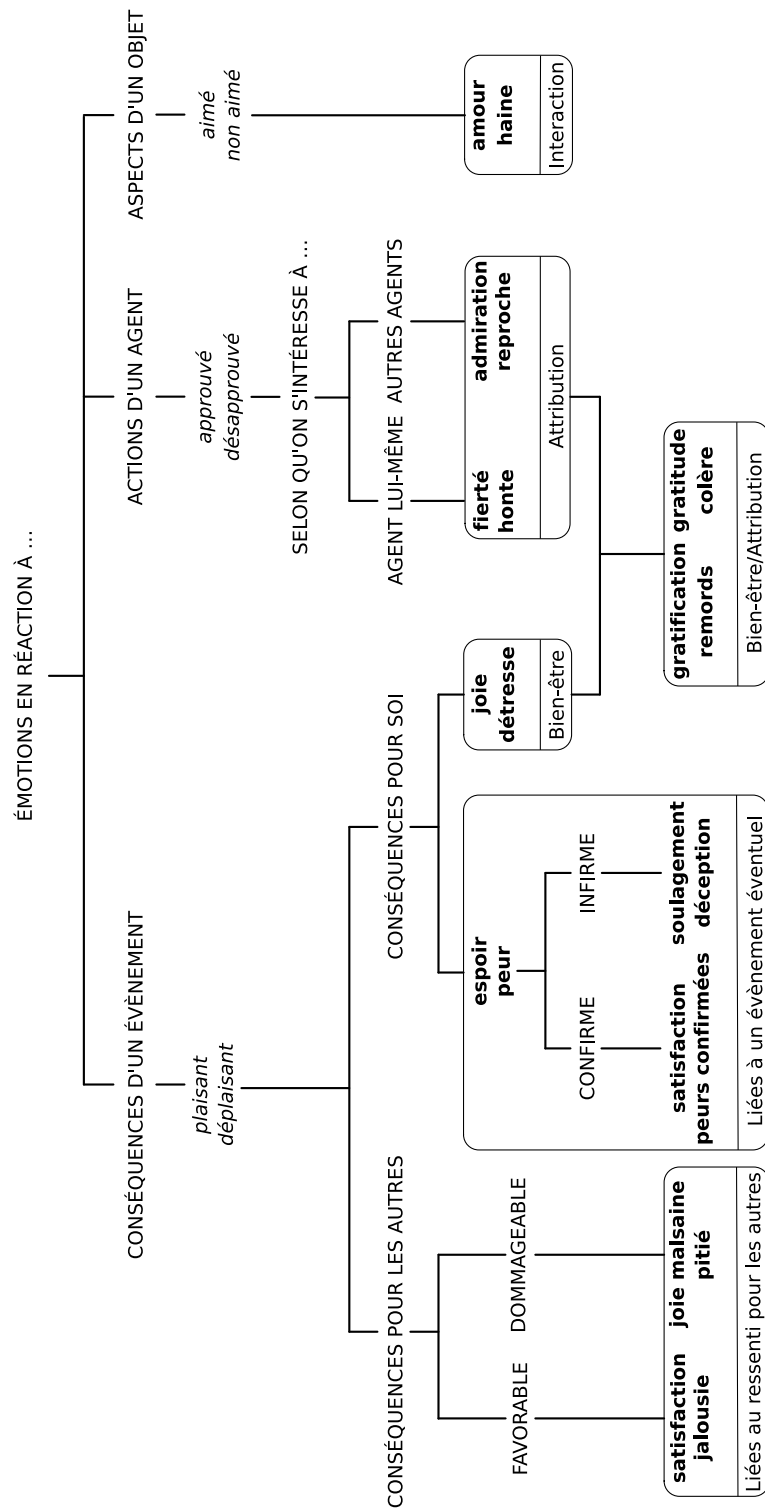


FIG. 1.8 – Structure globale des types d'émotions (Ortony et al., 1988).

donc dépendre des interactions prises en compte.

1.5 Humeurs et tempérament

On distingue les émotions d'une personne de ses humeurs ou encore de son tempérament. En effet, bien qu'étroitement liées, ces notions sont bien distinctes. Pour Goleman (1997), on aurait une structure concentrique autour des émotions :

À la périphérie se trouvent les humeurs, qui (...) sont moins vives et durent plus longtemps que les réactions émotives proprement dites (s'il est relativement rare d'être en proie à une violente colère un jour entier, il l'est moins d'être d'une humeur irritable, grincheuse, au cours de laquelle de courts accès de colère se déclenchent aisément). Au-delà des humeurs, on en arrive aux tempéraments, la propension à susciter une émotion ou une humeur donnée, qui rend les gens mélancoliques, timides ou gais.

(Goleman, 1997)

Le tempérament a donc une influence forte sur le processus émotionnel. En effet, une personne extravertie a tendance à ressentir plus souvent la joie qu'une personne introvertie et à l'exprimer plus facilement, par exemple.

1.6 Les sentiments

Ceci peut paraître assez surprenant mais la notion de sentiment a reçu beaucoup moins d'attention que celle d'émotion dans la recherche que ce soit en psychologie ou en sociologie. Une première raison à cela nous apparaît comme linguistique. La langue anglaise décrit avec le même mot les *sensations* et les *sentiments* (*feelings*). On parle d'ailleurs fréquemment d'*emotional feeling* dans les publications, mais nous allons y revenir. Il existe bien la notion de relations (*relationship*) avec la distinction entre *interpersonal relationship* (relations humaines) et *intimate relationship* (relations intimes), mais cela traduit plus la notion de relations générales entre personnes que celles-ci soient d'origine affectives ou sociales (professionnelles, de voisinage, ...). La langue française au contraire distingue clairement les *sensations*, les *sentiments* et les *relations*. Cette différence n'est pas réduite aux langues françaises et anglaises, mais traduisent au contraire une différence profonde entre les cultures anglo-saxonnes et latines (on retrouve ces caractéristiques entre les langues nordiques (Suédois, Finnois, Allemand) et les langues latines (Espagnol, Italien, ...)). Il nous a été difficile d'étendre cette comparaison à des cultures non européennes (africaines ou asiatiques notamment), mais il est probable qu'on y trouve également des approches très différentes.

La deuxième raison est un prolongement de la raison linguistique, puisque que dans la psychologie anglophone *feeling* décrit l'*expérience subjective consciente d'une émotion* (VandenBos, 2006). On parle donc comme dit précédemment d'*emotional feeling*. Clore and Gasper (2000) considèrent dans leur publication les deux termes comme synonymes. En Anglais donc,

les notions d'émotion et de sentiment sont donc intimement mêlées. On parle aussi en Français de sentiment de peur par exemple pour décrire la sensation d'une émotion à un instant donné, mais il est également usuel d'utiliser le mot sentiment pour décrire un état affectif construit sur la durée (amitié/inimitié, confiance/méfiance, ...).

1.7 L'influence de l'environnement

Nous allons aborder dans cette partie la notion d'environnement et l'impact que celui-ci peut avoir sur l'individu. L'environnement, comme nous le verrons dans le chapitre 2.1.3 consacré aux agents, se scinde en deux catégories : le produit global des stimuli dégagés par l'environnement physique dans lequel un individu évolue, on parlera de l'ambiance environnementale et des facteurs d'ambiance ; les stimuli perçus au cours des interactions avec d'autres individus du système, on parlera d'interactions sociales.

1.7.1 Les facteurs d'ambiance

Tout organisme vivant a des capacités d'adaptation à son environnement mais celles-ci sont coûteuses physiologiquement et cognitivement. La psychologie du travail s'intéresse aux effets de l'adaptation à l'ambiance sur les processus affectifs et cognitifs. Nous allons présenter les grandes recommandations qui ressortent de ces travaux (Gaillard, 1997).

L'ambiance lumineuse

L'ambiance lumineuse se décline selon deux axes : le niveau d'éclairement et l'effet de la couleur. Il est toujours souhaitable d'apporter un niveau d'éclairement suffisant dans toutes tâches sollicitant la vision, mais plus l'exigence de précision est importante, plus le niveau de lumière devra être élevé. Grandjean (1985) indique les valeurs recommandées. Cependant, il faut se méfier car lorsque le niveau d'éclairement est très élevé, le risque de contrastes trop violents devient important et peut être une cause de fatigue visuelle.

En ce qui concerne la perception des couleurs, le premier effet visuel est le pourcentage de réflexion qu'elle permet. Ce paramètre est à prendre en compte dans l'évaluation du niveau d'éclairement. Le second effet des couleurs est cognitif quand elles sont utilisées comme moyen de codage (bandes jaunes et noires pour risque de chute ou collision dans le domaine de la manutention). Enfin, les couleurs peuvent déclencher des effets subjectifs et émotionnels. Tout d'abord, il existe une association entre la couleur et l'impression thermique qu'elle évoque (le bleu est une couleur froide, par exemple). Ensuite, la couleur est liée à l'impression de repos/stimulation (vert, bleu et violet sont jugées reposantes alors que jaune et orange plutôt stimulantes et le rouge, c'est la couleur de la violence). Cependant, ces associations sont à prendre avec précaution car elles changent avec les goûts ou encore la culture.

L'ambiance sonore

Contrairement aux couleurs qui ont des effets très subjectifs, le bruit est facilement mesurable et ses effets bien cernés. Le seuil maximal d'audibilité est fixé à 80 dB. Au-delà de 100, on parle de douleur et si on dépasse 130 dB, le son est jugé intolérable. Des courbes de risque traumatique ont été établies par Wisner en 1987 [GAI 97]. Le risque traumatique est clairement lié à la durée d'exposition. En effet, de courtes durées d'exposition à des bruits d'intensité élevée provoquent une fatigue auditive suivie d'une récupération. Cependant, dans le cas d'expositions répétées, la fatigue auditive peut se transformer en surdité progressive irréversible. En plus des lésions physiques, le bruit a des effets dévastateurs sur la qualité de vie et le sommeil.

Sur le plan cognitif, les bruits porteurs d'information (conversation, téléphone) sont beaucoup plus perturbateur que les bruits n'ayant aucune signification particulière (bruits extérieurs, allées et venues). Il y a concurrence cognitive entre la tâche et la source du bruit. Dans certains cas, le bruit peut même devenir un stimulant, en particulier pour les activités répétitives et ennuyeuses. Cependant, il ne faut en aucun cas considérer que le maintien de vigilance peut être obtenu par le bruit.

L'ambiance thermique

Comme pour le niveau d'éclairement, Grandjean (1985) a établi des températures moyennes recommandées selon le type d'activité (mentale, manuelle assise ou sédentaire, ...). Pour avoir cette température en permanence, on a besoin de moyen de réchauffer ou rafraîchir la pièce. Pour cela, il faut prendre en compte la température effective (température ressentie en fonction de l'humidité et de la vitesse de l'air).

TAB. 1.3 – Température au travail selon Grandjean (1985).

Type d'activité	Température idéale en °C
Travail mental sédentaire	21
Travail mental léger, assis ou debout	18-19
Travail manuel pénible, debout	17
Travail très pénible	15-16

Le corps accepte plus ou moins bien les variations de température. En ambiance chaude, pour maintenir un même niveau de performance physique, les pertes hydriques par sudation doivent être compensées par un apport de volume équivalent, en petites quantités à chaque prise. Par contre, en ambiance froide, les activités manuelles sont plus perturbées pour deux raisons. Si le refroidissement affecte les membres, la dextérité est réduite et s'il atteint l'ensemble du corps, il provoque des tremblements. On peut ajouter que travailler dans un environnement où la température est inappropriée augmente le stress lié à l'hostilité du milieu.

1.7.2 Les interactions sociales

Les matrices d'interaction sociale

Les matrices d'interaction sociale ou MIS sont un moyen de modéliser les aspects sociaux des SMA. Sous ce terme générique, on regroupera un ensemble de notations matricielles adoptées pour décrire les liens qui peuvent unir des agents selon différents critères. Ainsi, Lechilli and Chaib-Draa (1996) utilisent une notation matricielle pour décrire ce qu'ils nomment la structure relationnelle entre agents. D'autre part, ce type de notation, introduit pour décrire les aspects sociaux d'un groupe d'individus, peuvent être transposés aux SMA comme on a pu le voir dans le chapitre 2.3.3 sur la notion de dépendance. Il est ainsi possible de décrire des dynamiques de groupe telle que la répulsion ou l'attraction. Il est également envisageable d'utiliser ce type de formalisme pour décrire les relations sentimentales ou émotionnelles que peuvent entretenir les uns vis-à-vis des autres.

Le modèle PROJEQUIP

Le modèle PROJEQUIP permet de comprendre la dynamique interne d'un groupe et de simuler de multiples configurations afin de valider des hypothèses de fonctionnement. Il est basé sur le modèle du « champ de force » établi par Lewin, le père de la dynamique des petits groupes, dans la première moitié du XX^{ème} siècle. Selon lui, « un groupe est un tout dont les propriétés sont différentes de la somme des parties ». Il explique alors l'action individuelle à partir de la structure qui s'établit entre le sujet et son environnement à un moment donné. Cette structure est un champ dynamique, c'est à dire un système de forces en équilibre. PROJEQUIP se base sur les résultats du questionnaire Echo. La mise en relation des scores de chaque individu sur les dix traits de personnalité permet d'obtenir des matrices caractérisant les forces qui peuvent exister entre deux personnes, mais aussi entre une personne et un groupe.

Les forces résultant du croisement des personnalités sont organisées en trois systèmes :

- les énergies donnent les forces attraction/répulsion,
- les motivations donnent les forces sympathie/antipathie,
- les adaptations donnent les forces synergie/antagonisme.

Structure relationnelle

Il existe au sein d'un groupe d'individu de multiple contraintes relationnelles. Ces contraintes peuvent être liés à des causes très différentes : rapports hiérarchiques, différence d'âge, de sexe, de classe sociale, dépendance dans le travail (collaboration ou concurrence), etc. L'ensemble de ces règles joue un rôle déterminant dans les interactions sociales entre individus.

Les notions de concurrence et de collaboration

La notion de concurrence est inhérente à la notion d'agent. L'autonomie de l'agent lui permet de gérer seul ces objectifs. Dans un système SMA, cela induit qu'il peut y avoir autant d'objectifs différents et donc potentiellement concurrents qu'il y a d'agents. La notion de concurrence apparaît dès lors que les agents partagent des ressources. Lechilli et Chaib-Draa ont présenté une classification des relations entre agents qui synthétise et étend les travaux précédents sur le sujet. Cette classification fournit également une notation matricielle permettant par exemple de détecter les éventuels conflits.

Conclusion

Nous avons décrit dans cette partie les principaux travaux et modèles à la base de la psychologie comportementale. Au niveau de la modélisation du comportement, et plus spécifiquement de l'étude des caractères propres à chaque individu, la théorie des traits et plus spécifiquement celles des *Big Five* est la plus plébiscitée et la mieux adaptée à une utilisation informatique car elle quantifie précisément un nombre de caractéristiques prédéfinies qui détermine les orientations comportementales de l'individu. L'extension du modèle original du *Big Five* par la société PerformanSe en un modèle bipolaire de 10 couples de traits complémentaires a été notre principal objet d'intérêt d'une part parce qu'il s'agit de la société où nous avons mené nos travaux et d'autre part parce que ce modèle s'avère à l'heure actuel l'un des plus éprouvé tant dans les domaines universitaires que professionnels. Pour ce qui touche à l'affect et plus particulièrement aux émotions, le modèle OCC déclinant 11 couples d'émotions elles aussi complémentaires est particulièrement bien adapté à la modélisation informatique car conçu à cette fin. Il est d'autre part largement utilisé. Enfin, nous avons indiqué des axes d'étude intéressants pour ce qui concerne les interactions entre individus et leurs impacts sur le comportement. Nous montrons dans le chapitre 4 l'utilisation que nous avons faite de ces travaux dans notre modélisation d'un agent émotionnel.

Les systèmes multi-agents

Ce chapitre est consacré aux systèmes multi-agents (SMA), et plus particulièrement à l'utilisation des SMA dans le contexte de la simulation comportementale. Les aspects individuel (entité agent) et collectif (interaction entre agents) y sont développés.

Sommaire

2.1	Principes et concepts	36
2.1.1	Historique	36
2.1.2	Définition	41
2.1.3	L'environnement	49
2.2	Les modèles d'agents	51
2.2.1	La problématique de la modélisation agent	51
2.2.2	L'agent non-délibératif	53
2.2.3	L'agent délibératif	55
2.2.4	Les agents hybrides	57
2.2.5	L'agent émotionnel	58
2.3	Les interactions entre agents	58
2.3.1	Les interactions entre agents	60
2.3.2	La communication entre agents	61
2.3.3	La notion de concurrence	62
2.3.4	Les langages de communication entre agents	63

Introduction

Produit hybride et multi-disciplinaire des recherches en intelligence artificielle, intelligence distribuée, robotique et psychologie, les systèmes multi-agents ou SMA sont le fruit des travaux afin de créer des entités virtuelles autonomes. Traduction de la volonté des chercheurs depuis les tous premiers temps de l'informatique de doter des systèmes de capacités propres à l'Homme,

les SMAs ont ainsi hérité des nombreux travaux sur l'apprentissage, le langage, la planification, la résolution de problèmes collaborative, ... Domaine à la confluence de l'informatique et des sciences humaines et naturelles, ils se sont également nourris des recherches en psychologie, en cognitive, en ethnologie, en sociologie, en biologie, ... Aujourd'hui matures, c'est une technologie largement utilisée du domaine ludique où on les utilise pour la construction d'entités virtuelles au comportement convaincant à des domaines hautement technologiques tels que les robots d'exploration planétaire.

Nos travaux concernant la modélisation d'une entité virtuelle au comportement affectif crédible ne pouvaient donc que s'inscrire dans ce domaine de l'informatique. Nous présenterons donc dans cette partie les principes et concepts de système multi-agents en retraçant l'histoire et en y présentant les différentes définitions que l'on peut trouver dans la littérature. Nous verrons dans un deuxième temps les modèles d'agents usuellement utilisés dans les travaux de recherche. En dernier lieu, nous traiterons de l'aspect fondamental que sont les interactions entre agents d'un même système ou de systèmes hétérogènes avec les normes de communication établies à cette fin.

2.1 Principes et concepts

2.1.1 Historique

Origines de la notion d'agent

Nous allons dresser ici un historique de l'émergence et de l'évolution du modèle multi-agents. Nous nous sommes appuyés sur plusieurs sources pour construire cette partie, la principale étant (Wooldridge, 2001). Michael Wooldridge est en effet à notre connaissance un des seuls à avoir réalisé un historique approfondi du domaine en appendice de son livre. Notre but est d'esquisser un panorama synthétique des personnes, travaux et principales avancées à l'origine du développement des systèmes multi-agents tels qu'on les connaît aujourd'hui. Les SMA sont un domaine multidisciplinaire qui puise ses origines dans de très nombreux domaines non limités à l'informatique pure, les sciences humaines ayant apporté beaucoup. On trouve ainsi des références dans l'intelligence artificielle, les systèmes concurrents, en passant par la théorie des jeux, les logiques, les langages ou encore la sociologie, les sciences cognitives. Il y a en réalité peu de domaines dans la recherche informatique se trouvant autant à la croisée de plusieurs disciplines, même si les SMA sont assez avant-coureurs d'une certaine transversalisation de l'informatique actuelle. La multi-disciplinarité du domaine rend son étude complexe, les références au sein des publications s'inscrivent en effet dans cette tendance en visitant et empruntant à des travaux nombreux et variés. Il serait donc illusoire et prétentieux de vouloir ici dresser un paysage exhaustif des systèmes multi-agents. Nous essaierons plutôt de mettre en évidence les publications clés du domaine et les principaux courants qui l'animent, cela afin de mieux cerner les principaux enjeux et pistes de recherche du moment.

La cybernétique ou l'âge des utopies

Au cours de la seconde guerre mondiale, les premiers ordinateurs analogiques ont été développés pour, la plupart du temps, assurer des tâches : automatisme avec la régulation de systèmes (contrôle des batteries anti-aériennes) et simulation basique de systèmes physiques (contrôle des flottes aériennes). Au sortir de cette guerre, un certain nombre de chercheurs, Norbert Wiener, Ross Ashby, et Grey Walter, découvrent au sein de ces systèmes des similitudes troublantes avec des processus biologiques comme par exemple la régulation de la température interne chez l'homme. L'idée germe alors d'utiliser ces machines pour construire les premiers être artificiels (Moravec, 1988). C'est le début de ce qu'on appellera la cybernétique¹⁰. L'objectif alors est d'unifier un certain nombre de recherches sur les mécanismes de contrôle animaux et humains au sein de modèles mathématiques exploitables dans les ordinateurs. Face aux nombreux obstacles tant au niveau des moyens techniques limités de l'époque que des difficultés à modéliser le vivant, la cybernétique déclina rapidement sans avoir pu produire les bases réellement nécessaires aux objectifs initialement poursuivis. Ces objectifs sont pourtant ceux qui aujourd'hui encore sous-tendent la plupart des développements qui s'inscrivent maintenant sous l'égide des systèmes multi-agents.

L'émergence de l'intelligence artificielle

Dans les années 60, avec l'avènement des premiers systèmes informatiques importants, une nouvelle discipline animée par un nouveau groupe de chercheurs voit le jour : l'intelligence artificielle qui elle va poser les bases nécessaires au développement ultérieur des SMA avec laquelle ils partagent une importante filiation. Les recherches prennent alors un tournant plus rationnel que celui plus biologique poursuivi par la cybernétique. On entrevoit en effet la possibilité de doter les machines de capacités de raisonnement leur permettant d'approcher certains mécanismes de déduction humains. C'est John McCarthy qui le premier en 1956 emploie le terme d'intelligence artificielle. L'agent intelligent n'a alors rien à voir avec sa définition actuelle. Il s'agit d'un imposant système dont la fonction est la résolution de problèmes bien spécifiques : la planification de tâches pour aboutir à un objectif, l'apprentissage par un programme informatique ou la communication entre différentes entités informatiques. Plusieurs raisons expliquent ce phénomène. Premièrement, la jeune discipline informatique se concentrait sur des problèmes de recherche au périmètre bien défini. Deuxièmement, il n'y avait pas à l'époque une volonté forte d'unification et d'intégration de ces différents domaines de recherche et de leurs développements liés en une seule entité. Enfin, cette volonté même n'aurait pu aboutir à cause des limitations techniques dues aux systèmes informatiques dont les capacités de stockage et les puissances de calcul restaient très limitées. Néanmoins, c'est à cette période que se dessinèrent les premiers piliers qui composent aujourd'hui les SMA. En effet, le travail de modélisation et de formalisation accompli durant cette période, notamment avec l'utilisation des logiques et les nombreuses idées empruntées aux sciences humaines est encore aujourd'hui à la base du développement de nombreux SMA.

¹⁰Du grec *kubernetes* qui signifie gouverner

En 1950, Alan Turing définit le test qui porte son nom (Turing, 1950) et qui sert à évaluer la proximité du comportement d'un ordinateur par rapport à celui d'un humain. C'était alors la première initiative visant à donner au terme intelligence artificielle une définition plus quantifiable et moins abstraite, l'idée étant de comparer en aveugle les réponses d'un ordinateur et d'un humain à des questions et d'essayer de déterminer qui est qui. Même si certaines critiques ont tenté vainement d'affaiblir sa valeur, notamment Gedankenexperiment¹¹ de Searle (Searle, 1980), il est encore reconnu comme une mesure de référence aujourd'hui. Herbert Simon et Allan Newell créent la sensation avec The Logic Theorist capable de démontrer trente huit des cinquante deux premiers théorèmes du deuxième chapitre des Principia Mathematica de Russell et Whitehead. C'est alors pour la communauté le premier programme présentant un comportement considéré à l'époque comme intelligent. En 1958, John McCarthy crée le langage LISP (LISt Processing), langage procédural qui connut et connaît encore une forte utilisation dans l'élaboration de systèmes de raisonnement. Avec FORTRAN, il fut l'un des langages utilisés pour coder les premiers systèmes experts, qui peuvent être considérés comme les premiers programmes capables de résoudre des problèmes complexes. Les principales recherches à l'époque visaient donc à développer des langages et des systèmes aptes à la résolution de problèmes complexes, plutôt que des entités autonomes et intelligentes. L'intelligence artificielle se situait donc plutôt au niveau des langages et de l'ordinateur dans son ensemble.

La robotique et l'intelligence artificielle distribuée

En 1950, Turing voyait deux manières de concevoir une entité intelligente. La première voie consiste à développer des systèmes capables de reproduire des activités intellectuelles humaines telle que jouer aux échecs ou traduire un texte. C'est celle suivie par l'intelligence artificielle, Brooks l'a nommée la voie désincarnée¹² (Brooks, 1991). La seconde voie consiste « à doter une machine avec les meilleurs organes sensitifs qu'on puisse acheter, et ensuite lui apprendre à comprendre et parler anglais »¹³. C'est celle suivie par la robotique à partir du milieu des années 60, un domaine dérivé de l'IA classique. On peut voir dans ces deux domaines la tête et les jambes de la machine intelligente. En effet, si d'un côté l'essentiel du travail s'est porté sur l'élaboration de systèmes aptes à mettre en oeuvre des raisonnements. De l'autre, le but a été de créer des robots capables à partir de données fournies par des capteurs d'accomplir certaines actions coordonnées. La robotique a permis l'élaboration de robots et d'automates complexes tels ceux utilisés dans l'industrie. Tout comme dans le domaine de l'IA, ces robots n'ont rien d'autonomes ou d'adaptatifs à des environnements changeant sur lesquels ils pourraient extraire et manipuler de l'information.

Parallèlement, alors que l'IA et la robotique se concentrent sur une entité individuelle, un second champ de recherche spécifique s'est développé : l'intelligence artificielle distribuée, attachée à la solution de problèmes de concurrence et de distribution dans les problèmes d'IA

¹¹Expérience de pensée : démarche qui consiste à imaginer une situation concrète, mais fictive, qui met en jeu une notion abstraite (ici la compréhension), et à observer cette situation et se demander « que se passe-t-il si ... » ?

¹²*Unembodied path* en anglais

¹³« To provide the machine with the best sense organs that money can buy, and then teach it to understand and speak English. »(Turing, 1950)

classique. Le but principal est de mutualiser les ressources pour résoudre des problèmes plus complexes en bénéficiant d'une puissance de calcul plus élevée. A la différence des multi-agents tels que nous les définirons par la suite, les agents ici utilisés sont coopérants, omniscients et aux ressources illimitées, évoluant dans des environnements dénués de conflits avec des tâches bien spécifiées. On verra pourtant que c'est sans doute le domaine qui se rapproche le plus des systèmes multi-agents.

La scission IA classique / vie artificielle

L'IA a donc pendant longtemps concentré ses efforts sur l'utilisation de logiques descriptives et de représentations symboliques pour parvenir à ses fins : une entité unique assurant l'intégralité des traitements. Ces méthodes lourdes présentaient l'avantage d'être relativement prédictibles, mais nécessitaient de gros effort de conception et de paramétrage. Avec l'évolution de l'informatique, vint l'idée de coupler plusieurs de ces méthodes, avec à terme l'ambition de créer une entité autonome et intelligente capable de traiter des problèmes multiples, voir de communiquer avec d'autres entités du même type. C'est au milieu des années 80 qu'une autre approche émergea avec le sentiment qu'il n'était pas possible de construire un agent avec ces outils. Cette nouvelle communauté, aujourd'hui nommée vie artificielle ou encore intelligence artificielle située, avançait l'idée qu'un comportement intelligent pouvait émerger de l'interaction de comportements plus simples sans faire appel aux méthodes lourdes développées en IA classique. Ce n'est plus une approche analytique mais systémique ou holistique qui entre en jeu, considérant que les modèles destinés à créer des entités intelligentes ne peuvent pas faire l'économie du contexte dans lequel évolue l'entité et des interactions qu'elle entretient avec lui. L'environnement devient essentiel et avec lui le déterminisme des premiers systèmes s'envole.

Rodney Brooks a été un des fers de lance de cette nouvelle vague (Brooks, 1986) qu'on a aussi appelée IA comportementale ou IA réactive¹⁴. Il apparaissait en effet que les techniques développées jusqu'alors en IA restaient très théoriques et rendaient impossible la réalisation concrète d'agents intelligents. L'approche plus pragmatique de l'IA comportementale va rapidement donner des résultats encourageants avec entre autres la conception des premiers robots capables d'une certaine forme d'autonomie. Car c'est là le nouveau critère avancé par les tenants de cette nouvelle vague : l'intelligence se mesure à son autonomie et à sa capacité à s'adapter à son environnement et ses variations. Le système n'est donc plus figé dans un raisonnement pré-déterminé, mais évolue au gré de ses interactions. Une séparation assez franche apparaît donc alors entre les deux communautés : IA classique et vie artificielle, cette dernière s'inspirant plutôt des modèles biologiques. Avec cette approche, la nouvelle communauté va élargir ses champs d'investigation à de nouveaux domaines très divers des sciences : la biologie, la cognition, l'ethnologie. En plus de contribuer à un certain renouveau des idées, cette incursion dans de nouveaux domaines, couplée aux sciences dures déjà très présentes en informatique : mathématiques, statistiques, physique, va être avant-coureuse du métissage et de l'interdisciplinarité de l'informatique actuelle.

¹⁴Behavioral AI ou Reactive AI en anglais

La communauté multi-agents

La communauté agent se situe au croisement de l'ensemble de ces communautés et puise ses idées à la fois dans les méthodes de raisonnement purement logiques de l'IA classique, dans les travaux de gestion de processus distribués de l'IAD, dans l'approche comportementale de la vie artificielle et, pour la branche des agents physiques, pour une bonne part dans la robotique pour les mécanismes de perception de l'environnement et de contrôle des automates. A la différence de l'ensemble de ces domaines, l'idée d'autonomie occupe une place prépondérante dans la communauté agents, même si l'agent et son comportement restent le plus souvent sous une forte supervision humaine. Les interactions entre entités occupent également une large part avec notamment des efforts importants sur la standardisation et la normalisation de langages de communication et l'étude des mécanismes de coopération et concurrence qui emprunte beaucoup aux sciences humaines (économie et sociologie) et à l'informatique distribuée. Sur ces 2 points, la communauté SMA est très proche de la vie artificielle. Là où elle s'en différencie ou plutôt où elle évolue par rapport à cette dernière, c'est dans la conception interne de l'agent qui va s'enrichir pour le rendre capable d'exécuter des tâches de plus en plus complexes. C'est là qu'elle se rapproche de l'IA classique à laquelle elle va emprunter bon nombre de travaux développés dans les années 60-70. La planification des tâches effectuées par l'agent sera une des clés de voûte du développement des agents et est directement issue de ces travaux. Les techniques d'apprentissage sont également très employées dans certains systèmes. Selon Innes Ferguson (Ferguson, 1992), il est nécessaire de combiner ces deux domaines afin de parvenir à un agent satisfaisant.

En réalité, jusque dans les années 90, la communauté s'occupant des systèmes multi-agents et celle travaillant sur l'agent individuel sont restées assez indépendantes. Pourtant, les premiers systèmes pouvant être considérés comme à l'origine des SMA sans en revendiquer le titre sont apparus dans les années 70. Tout d'abord, on peut citer les blackboard systems où plusieurs entités indépendantes utilisaient un « tableau noir » pour communiquer (système Hearsay (Herman et al., 1980)) et décrits en 1962 par Alan Newell (Newell, 1962). Parallèlement, d'autres systèmes comme BEINGS de Doug Lenat (Lenat, 1975) ou ACTOR de Carl Hewitt (Hewitt et al., 1973) ont été conçus sur un principe de communication entre différents agents. Ces systèmes sont encore aujourd'hui reconnus comme porteurs de beaucoup d'idées fondamentales des SMA, même si la recherche sur les blackboard systems est un peu tombée en désuétude. Vers la fin des années 70, Reid Smith conçoit à l'Université de Stanford le système Contract Net. Ces développements accompagnaient le pressentiment de plus en plus présent dans la communauté informatique, et devenu réel avec l'apparition d'Internet, que l'avenir de l'informatique se ferait au travers des systèmes distribués et ouverts.

C'est dans les années 80 que les SMA ont connu une véritable progression avec l'apparition de forums scientifiques de discussion dédiés au domaine et de la première conférence sur l'IA distribuée DAI initiée par Randall Davis en 1980. Les travaux sur les multi-agents deviennent alors très nombreux. Huhns (1987) est à l'origine du premier livre considéré comme la référence sur le sujet, suivi par un deuxième volume (Gasser and Huhns, 1989) écrit en collaboration avec Les Gasser, à l'origine notamment du système MACE (Gasser et al., 1987). En 88, Gasser publie avec Alan Bond, un ouvrage de synthèse regroupant de nombreuses publications Readings

in Distributed Artificial Intelligence (Bond and Gasser, 1988). Parmi d'autres auteurs de références, on peut citer Bratman (1987), Maes (1991), Agre and Chapman (1987), Wooldridge and Jennings (1995b), citetRosenschein.Genesereth.1987, Shoham (1993). On peut citer également au Japon, la conférence MAAC et en Europe, le projet MAGMA (Modelling an Autonomous aGent in a MultiAgent world) en 89, ainsi que la conférence MAAMAW également créée en 89 et dirigée depuis par Yves Demazeau. En France, Jacques Ferber a publié en 95 un ouvrage de référence sur les multi-agents (Ferber, 1995), et depuis 1993 ont lieu les JFIADSMA (Journées Francophones d'Intelligence Artificielle et De Systèmes Multi-Agents).

Aujourd'hui

Les systèmes multi-agents continuent comme l'ensemble des domaines de l'informatique à bénéficier de l'augmentation toujours soutenue des capacités de calcul des machines actuelles. Après l'informatique répartie¹⁵, le phénomène plus récent de l'augmentation massive de l'interconnectivité globale de l'ensemble des systèmes contribue également à un renouvellement des domaines de recherche en sonnant l'avènement de l'informatique diffuse¹⁶. La troisième dimension est l'augmentation considérable du volume de données et du nombre de leurs sources. De nouveaux axes de recherches dans cet environnement fortement dynamique et hétérogène apparaissent donc.

Dans un premier temps, on assiste à un effort d'homogénéisation dans le domaine de la modélisation SMA avec une initiative comme AgentUML (Bauer et al., 2001) à l'origine de nombreuses propositions pour la modélisation des agents largement reprises dans la version 2.1.2 d'UML (OMG, 2007). La FIPA, à l'origine d'AgentUML, est aussi à la base de nombreux travaux de normalisation des interactions entre agents. Les spécifications concernant les modèles de communication agents sont le cœur de leurs travaux (The Foundation for Intelligent Physical Agents, 2002), mais ceux-ci s'étendent sur plusieurs domaines : le transport des messages des messages, les protocoles d'interactions entre agents, les performatives de communication, le contrôle des agents au sein des plates-formes multi-agents, les ontologies de communication, le déploiement des agents, etc. Les environnements et plates-formes orientés multi-agents se sont aussi multipliés : JADE et Jadex Bellifemine et al. (2007), AgentBuilder, MadKit, etc.

2.1.2 Définition

Il nous est nécessaire pour la suite de définir la notion d'agent telle que nous l'entendons. Comme nous venons de le voir, les systèmes multi-agents et plus spécifiquement l'agent puisent leur origine dans de nombreux domaines. Cette interdisciplinarité tend à rendre difficile l'établissement d'une définition unanimement partagée d'un agent. En effet, chacun des travaux effectués affichent le plus souvent une filiation plus prononcée avec tel ou tel domaine originel, la plupart du temps parce que le chercheur ou groupe de chercheurs en est issu. De ce fait, les

¹⁵Grid computing, en anglais.

¹⁶Pervasive computing, en anglais.

objectifs poursuivis dans la création de l'agent peuvent être très divers à l'image de la multiplicité de ses origines. On ne trouve donc pas dans la littérature de consensus sur la définition d'un agent. Néanmoins, certains chercheurs ont tenté de caractériser ce métissage en déterminant les particularités partagées par les différents agents que l'on peut rencontrer. Nous ne nous orienterons pas vers une énumération des différentes définitions existantes¹⁷, mais tenterons plutôt d'établir au travers de quelques auteurs une image correcte et assez précise de la notion d'agent. Les principales caractéristiques développées ici seront reprises et approfondies dans la partie 2.2.

La définition usuelle

On peut, dans un premier temps, considérer la définition très synthétique qui nous est proposée par le dictionnaire :

AGENT n.m. (du latin scolastique *agens*, participe passé de *agere*, agir) Ce qui exerce une action, par opposition à ce qui la subit.

(Dictionnaire de l'Académie Française)

Cette définition succincte souligne les deux caractéristiques principales d'un agent. L'agent est une entité capable d'agir. On verra par la suite que cette action peut se porter à la fois sur son environnement¹⁸ pour infléchir sur son évolution et également sur son état propre. Les actions externes peuvent revêtir un grand nombre de formes, de la simple communication au maniement d'un objet de l'environnement. Les actions internes sont également très nombreuses, de la mise à jour de ses connaissances internes à la décision de la tâche à accomplir.

La deuxième partie de la définition est la plus intéressante car elle porte la lumière sur ce qui distingue un agent d'une entité informatique classique. En effet, l'agent n'est en théorie pas ou peu censé subir d'interventions tendant à lui imposer ses actes. Ici émerge le critère fondamental de tout agent : l'autonomie. Là où la plupart des systèmes se bornent à exécuter des actions conformément aux instructions qui leur ont été passées, un agent possède, de manière plus ou moins prononcée, une forme de libre arbitre qui doit lui permettre de choisir l'action qui lui convient le mieux. Bien entendu, n'étant qu'une entité informatique, cette autonomie peut-être relativisée par la présence de l'homme dans la conception de l'ensemble des règles qui régissent l'agent.

Cela rappelle le débat sous-jacent de savoir si l'homme est en mesure de produire une entité suffisamment complexe pour que son comportement final lui échappe et puisse donc être considérée comme autonome. C'est donc de savoir si un ensemble de règles déterministes peuvent engendrer un comportement indéterministe. Cette question a souvent été débattue et est notamment à la base de la scission vie artificielle/IA classique que nous avons vue précédemment. C'est le point de vue de l'étude des systèmes complexes qui postule qu'un système constitué d'un grand nombre d'entités en interaction peut être considéré comme complexe lorsqu'un observateur ne peut pas prévoir de manière sûre l'évolution de ce système et ce malgré une

¹⁷Franklin et Graesser ont recensé un grand nombre de définitions dans leur taxonomie des agents (Franklin and Graesser, 1997).

¹⁸On verra ultérieurement dans la partie 2.1.3 ce qu'on entend par la notion d'environnement.

connaissance parfaite des éléments le constituant. Le travail ici exposé penche effectivement pour cette hypothèse puisque l'objectif est de produire un système d'entités interagissant dont il soit nécessaire d'étudier le comportement grâce à des méthodes de fouille de données afin de révéler les ressorts de l'émergence de certains événements.

Une définition détaillée et formelle selon Ferber

Si cette première définition nous permet de clarifier la nature profonde de ce qu'est un agent au sens large du terme, la communauté scientifique a néanmoins dégagé un certain nombre d'autres caractéristiques supplémentaires propres à un agent informatique. Elles apportent, en sus de la dimension du modèle informatique et des exigences qu'il doit rencontrer afin de satisfaire à cette contrainte d'autonomie, un apport sémantique important sur la définition d'un agent et de l'ensemble des caractéristiques induites lorsqu'on évoque le concept d'agent. Jacques Ferber a proposé une définition relativement complète et éloignée de tout particularisme propre à telle ou telle communauté (Ferber, 1995). Selon lui donc :

Un agent est une entité réelle ou virtuelle :

1. qui est capable d'agir dans un environnement,
2. qui peut communiquer directement avec d'autres agents,
3. qui est gouvernée par certaines tendances (sous la forme d'objectifs individuels ou d'une fonction de satisfaction, voire de survie, qu'elle cherche à optimiser),
4. qui possède des ressources propres,
5. qui est capable de percevoir (mais de manière limitée) son environnement,
6. qui ne dispose que d'une représentation partielle de cet environnement (et éventuellement aucune),
7. qui possède des compétences et offre des services,
8. qui peut éventuellement se reproduire,
9. dont le comportement tend à satisfaire ses objectifs, en tenant compte des ressources et des compétences dont elle dispose, et en fonction de sa perception, de ses représentations et des communications qu'elle reçoit.

((Ferber, 1995))

La définition de Ferber dessine le portrait d'un agent comme une entité informatique complexe aux multiples compétences. Une des réserves souvent exprimées vis-à-vis des agents était de savoir ce qui les différenciait de logiciels classiques et des objets que l'on manipule en programmation (Wooldridge, 2001). La définition de Ferber présente l'avantage de formuler clairement les caractéristiques d'un agent. Elle lève donc cette ambiguïté en clarifiant à la fois ce que l'agent possède et ce qu'il offre par rapport à un système informatique classique.

La première de ces qualités, la capacité d'agir dans un environnement, peut paraître abstraite. Elle définit en fait la nature de l'agent en tant qu'entité évoluant dans un environnement,

ou plus formellement l'agent comme une unité dont tout ce qui extérieur compose l'environnement. Cette propriété est également applicable à la plupart des composants informatiques, de la méthode au sein d'un objet à l'objet lui-même jusqu'au système logiciel dans son ensemble, qui peuvent être représentés comme des entités recevant des entrées provenant de sources extérieures : leur environnement, les traitant et restituant des sorties à ce même environnement. La capacité d'action de l'agent ici définit n'a pour seul objectif que de rappeler la nature la plus simple d'un agent telle qu'on a pu la voir dans la définition usuelle : une agent est une entité qui agit, qui plus est dans un environnement.

La communication évoquée ensuite est un point propre à l'agent et plus particulièrement aux systèmes multi-agents, et une première distinction fondamentale d'avec les systèmes informatiques classiques. Là où un logiciel n'a généralement aucune contrainte de communication avec d'autres logiciels¹⁹, l'agent doit être capable d'interagir avec les autres entités qui peuplent son environnement, cette interaction se manifestant le plus souvent par des flux de communication. Comme on le verra par la suite, la capacité de communication avec d'autres agents peut être grandement relativisée, tout dépend en fait de l'environnement dans lequel évolue l'agent. Dans un environnement hétérogène, comme l'est Internet aujourd'hui, peuplé d'entités de différentes natures et différentes origines, la contrainte de communication est lourde à mettre en place car elle implique la mise en place de standards de communication entre ces différentes entités. A contrario, dans un environnement homogène dont l'ensemble des entités sont issues du même moule, comme cela peut l'être dans une simulation, la communication se simplifie grandement. A plus forte échelle encore, un agent autonome, individu unique dans son environnement, pourra n'avoir comme seule contrainte de communication celle avec l'utilisateur du système. On voit donc bien que selon la complexité du système à développer et son intégration au sein d'un environnement lui-même plus ou moins complexe, la communication pourra être anecdotique ou au contraire fondamentale. Enfin, autre point abordé ici par Ferber, l'importance d'un canal direct de communication entre entités est plus à prendre en compte au niveau de l'autonomie de l'agent. En effet, d'autant plus dans les environnements virtuels, la communication entre agents reposera toujours sur des systèmes de communication et n'est donc que théoriquement directe. De plus, l'agent n'étant pas censé être omniscient (points 5 et 6 de la définition), on ne peut assurer ni que ses messages conserveront leur intégrité durant leur acheminement, ni que l'agent récepteur en saisira le sens initial, la notion de canal direct n'a donc sous cet angle là plus que peu d'importance.

Les points 3 et 9 de la définition sont complémentaires. Les tendances d'un agent sont un point discutable selon l'architecture d'agent que l'on choisit. En effet, comme on le verra dans la section 2.2, la notion de tendance si elle a le plus souvent un intérêt pour un agent délibératif, peut n'en avoir aucun pour un agent purement réactif. Plus encore, la notion d'objectif à satisfaire n'est pas présente ou sous une forme vraiment primaire dans les agents réactifs. Encore une fois, nous verrons que ces caractéristiques sont à relativiser selon l'environnement dans lequel on se place et les compétences dont on désire doter l'agent.

¹⁹Si ce n'est avec le système d'exploitation sur lequel il repose, mais cette communication est assurée par les bibliothèques sur lesquelles il se base, et le système d'exploitation peut être considéré comme l'environnement du logiciel plutôt qu'une autre entité.

La notion de ressources est une notion commune à tous les programmes informatiques dès lors que l'on s'intéresse à la concurrence. En effet, le temps processeur, les accès disque, les capacités mémoires et les débits réseaux sont parmi les ressources primordiales avec lesquelles doit composer tout processus informatique. Cette notion est encore plus présente dès lors qu'on parle de processus distribués et donc a fortiori d'IAD. Les systèmes multi-agents, également enfants de cette discipline, n'échappent donc pas à la règle et sont directement concernés par la gestion des ressources systèmes qui plus est dans environnement fortement concurrent où certaines contraintes fortes doivent être respectées. Dans certains systèmes, on ne peut en effet envisager que la répartition des ressources systèmes entre entités ne soient pas équitable sous peine de désavantager des entités par rapport à d'autres et donc d'en fausser les résultats. Cette gestion est souvent déléguée à la plate-forme SMA qui supporte l'exécution des agents, qui elle même se repose en dernière instance sur le système d'exploitation sur lequel elle tourne. D'autre part, les ressources de l'environnement prennent un sens plus large dans les systèmes multi-agents car elles ne se limitent pas aux ressources systèmes et peuvent aussi se rapporter à l'ensemble des ressources virtuelles dans un environnement dématérialisé, ou réelles pour des agents robots dans un environnement physique²⁰. Dans ces cas bien précis, la gestion des ressources doit être prise en charge au sein des agents et suppose donc des modèles fiables. Enfin, la ressource de l'agent peut être également vue comme les moyens à sa disposition pour satisfaire ses objectifs. En cas de ressources limitées ou à l'impact critique, l'agent doit alors chercher la meilleure adéquation entre l'utilisation de l'une d'elle et l'accomplissement d'un objectif. Cela est généralement accompli grâce à des algorithmes de planification, ou dans les systèmes avec apprentissage par un système essai-erreur qui peut, par exemple, diriger un réseau de neurones.

Les points 5 et 6 de la définition qui concernent le rapport de l'agent à son environnement tant au niveau de la perception que de la connaissance qu'il s'en représente insistent sur le fait qu'un agent réaliste ne doit pas être omniscient, ce qui se traduit le plus souvent par un accès limité au niveau perceptif aux informations disponibles dans l'environnement et une connaissance donc elle aussi limitée sur cet environnement. La limitation perceptive peut être due d'une part à un accès impossible ou partiel à certaines données, et d'autre part à une possible altération des données perçues par rapport aux données réelles (comme nous l'avons vu au sujet de la communication par exemple). Cette notion d'omniscience sera approfondie avec celle de l'omnipotence dans la section consacrée à l'environnement.

Les compétences d'un agent désignent ses connaissances sur le domaine. À partir de celles-ci, il va être capable d'offrir des services : éventail des actions qu'il est capable d'effectuer. Il est important de considérer ce critère du seul point de vue de l'agent qui agit dans un environnement. En effet, aujourd'hui, avec notamment les services Web, la notion de service est définie comme une fonction offerte par un système au monde extérieur. On ne peut pas adopter cette vision pour les agents, car elle peut être très réductrice. Dans le cas qui nous concerne d'une simulation par exemple, le système fonctionne en vase clos dans un but d'observation, les services ne peuvent donc être entendus que comme les actions que peuvent réaliser les agents. Les

²⁰Auquel cas, la notion de concurrence pour les ressources systèmes devient même secondaire puisque souvent l'entité dispose à elle seule d'un système entier.

compétences et services sont ainsi 2 notions très complémentaires à celle de rôle d'un agent : ensemble d'aptitudes et d'actions qu'il est capable de réaliser.

Nous ne détaillerons pas la notion de reproduction des agents qui n'est pas réellement une propriété significative et partagée par les agents. Comme indiquée, c'est une éventualité qui peut être envisagée au même titre que la capacité d'apprentissage de l'agent comme nous le verrons dans la suite.

Le dernier point de la définition est sans doute le plus important puisqu'il structure l'ensemble des caractéristiques précédentes en déterminant le principe de fonctionnement de l'agent. Sont ainsi clairement présentées les articulations entre l'ensemble des composants primordiaux du comportement de l'agent. L'agent agit selon ses perceptions et la représentation qu'il a du monde dans la limite des ressources et des compétences qu'il a sa disposition. L'ensemble de ses actes est dirigé vers un but. Comme nous le verrons dans le chapitre 2.2, la notion de but est absente ou adaptée dans certains modèles.

Une définition formelle de l'agent

On peut fournir la définition formelle suivante d'un agent :

Agent $N = (\mathcal{P}, \mathcal{C}, \mathcal{A}, p, f, u)$ où :

- $\mathcal{P} = \{p_1, p_2, \dots, p_i\}$: l'ensemble des perceptions de N ,
- $\mathcal{C} = \{c_1, c_2, \dots, c_j\}$: l'ensemble des connaissances de l'agent N ,
- $\mathcal{A} = \{a_1, a_2, \dots, a_k\}$: l'ensemble des actions réalisables par N ,
- $p : \mathcal{P}^{l*} \times \mathcal{C}^m \rightarrow \mathcal{A}^n$: la fonction des possibles de N ,
- $u : \mathcal{F}(\mathcal{P}^{l*} \times \mathcal{C}^m, \mathcal{A}) \rightarrow \mathbb{R}$: la fonction d'utilité de N ,
- $f : \mathcal{P}^{l*} \times \mathcal{C}^m \rightarrow \mathcal{A}$: la fonction de choix de N .

Cette définition est générique et nous verrons par la suite que certains types d'agent s'y conforment en y apportant des adaptations. À partir de ses perceptions et de ses connaissances, la fonction p détermine l'ensemble des actions possibles de l'agent. Une fonction d'utilité u (ou de performance) permet d'évaluer et d'ordonner les différentes actions possibles pour choisir la mieux adaptée au contexte. Enfin, la fonction de choix f détermine l'action réalisée par l'agent, en combinant la fonction des possibles p à la fonction d'utilité u . Les connaissances telles que nous les définissons ici sont à prendre au sens large, elles concernent aussi bien les croyances de l'agent sur son environnement que ses propres savoir sur son état interne, notamment ses buts. La fonction des possibles détermine donc les différentes actions possibles en fonction du contexte environnemental de l'agent et de son contexte interne. Nous verrons que c'est notamment au niveau de ces fonctions p et u que se situent les principales différences entre les grands modèles d'agent.

Une définition minimale et pragmatique selon Wooldridge et Jennings

Wooldridge et Jennings (Wooldridge and Jennings, 1995a) ont également proposé une définition intéressante de l'agent articulée autour de quatre concepts clés qui caractérisent un agent.

L'intérêt de leur définition est qu'elle restreint la définition de l'agent à l'adéquation avec ces quatre qualités.

Un *agent* est (...) [un] système informatique qui a les propriétés suivantes :

- autonomie : les agents opèrent sans l'intervention directe d'humains ou autres, et ont une certaine forme de contrôle sur leurs actions et leur état interne,
- sociabilité : les agents interagissent avec d'autres agents (et éventuellement des humains) via un type de langage de communication agent,
- réactivité : les agents perçoivent leur environnement, (qui peut être le monde physique, un utilisateur via une interface graphique, un ensemble d'autres agents, Internet ou même tout ceci combiné), et répondent dans un temps raisonnable afin de changer ce qui s'y passe,
- capacité d'initiative²¹ : les agents n'agissent pas simplement en réponse à leur environnement, ils sont capables de faire preuve d'un comportement dirigé vers un but en prenant l'initiative.

(Wooldridge and Jennings, 1995a)

Cette définition, bien que moins générale que celle de Ferber puisqu'elle se rapporte uniquement aux agents informatiques, a le mérite d'être beaucoup plus pragmatique en se rapprochant directement des problèmes de modélisation et d'implémentation des agents. Quatre propriétés sont définies qui peuvent directement servir de critères d'évaluation pour tout système multi-agents sur un abaque permettant de le mesurer par rapport à chacune de ces exigences. On peut donc plus facilement déterminer le degré d'adéquation d'un système avec l'agent ou le SMA idéal (selon Wooldridge et Jennings).

Chacune des propriétés est elle-même définie de manière très pragmatique. Ainsi, l'autonomie n'est pas un prérequis absolu. L'agent doit avoir *une certaine forme de contrôle sur [ses] actions et [son] état interne*. L'autonomie peut donc être plus ou moins développée, on pourra parler d'une autonomie modérée.

La sociabilité définit simplement la capacité d'interaction, et plus particulièrement de communication, des agents avec leur environnement, et introduit la notion d'ACL (Agent Communication Language). En effet, les interactions dans les SMA se manifestent intégralement via des primitives de communication (les performatives selon la théorie des Speech Acts (Searle, 1969)) qui peuvent ainsi traduire aussi bien des communications verbales telles qu'on peut en rencontrer entre humain, que des interactions physiques avec une partie de l'environnement (dans les environnements virtuels notamment). Cette particularité est liée au concept d'autonomie qui impose l'absence de mode d'action sur un agent autre que le passage de messages (ou stimuli).

La réactivité, et la notion de temps de réponse raisonnable, est une caractéristique très intéressante car souvent inconsciemment laissée de côté lors de la conception de SMA. Elle est pourtant fondamentale si on veut garantir d'une part l'efficacité du système et d'autre part l'autonomie de l'agent. En effet, la performance du système, sa capacité à percevoir correctement les variations de l'environnement et son temps de réaction à ces variations conditionnent sa ca-

²¹*pro-activeness*, en Anglais. Dans la littérature, ce critère est également appelé *goal-oriented* ou *purposeful*.

pacité à accomplir ses tâches en détectant les facteurs favorables ou défavorables par rapport à ses objectifs et pouvant survenir lors de modifications de l'environnement. Un agent dont le processus décisionnel est trop long ne peut donc prétendre répondre correctement à ce critère. On peut ainsi noter que l'embarquement au sein d'un agent de procédures lourdes d'intelligence artificielle peut nuire à sa capacité de réaction, et donc à son autonomie, en alourdissant son temps de traitement.

Enfin, la capacité d'initiative rejoint les points 2 et 9 de la définition de Ferber et appuie la notion d'autonomie en définissant la nature proactive de l'agent. Son comportement est gouverné par des ressorts internes. Les messages (ou instructions) reçus de l'environnement s'ils influent sur les actions de l'agent ne permettent pas de le commander de manière déterministe, une ou plusieurs contraintes internes (objectifs, buts, tendances, fonction de survie, etc.) orientant ses choix.

L'apprentissage

Une notion importante mais non évoquée dans les définitions de Ferber et Wooldridge et Jennings est la capacité d'apprentissage de l'agent. Il s'agit de sa capacité à faire évoluer son mode procédural grâce aux connaissances acquises dans son environnement. L'apprentissage est ainsi une notion complémentaire mais bien distincte de la mémorisation, notion largement plus répandue dans les SMA. Là où la mémorisation permet à un agent de stocker sa perception de l'environnement sous forme de connaissance, l'apprentissage traite de sa faculté à assimiler ses connaissances d'un point de vue procédural de sorte qu'elles lui permettent de modifier ses règles comportementales. Les travaux afférents à l'apprentissage dans les SMA sont largement connexes à ceux réalisés en intelligence artificielle dans le domaine de l'apprentissage automatique (*machine learning* en anglais). Là encore, le spectre des utilisations de l'apprentissage au sein des agents est très vaste. La première et la plus fréquente est l'utilisation du paradigme «essai-erreur» utilisé par exemple par un agent pour évaluer parmi une base de plans d'action les mieux adaptés aux problématiques qu'il rencontre dans son environnement. On peut trouver des systèmes plus évolués reposant sur des squelettes ou des briques de plans d'actions que l'agent va apprendre à assembler pour résoudre des problèmes. On voit ici une manière d'augmenter l'autonomie d'un agent en lui permettant de modifier dynamiquement son comportement. Les réseaux de neurones et réseaux bayésiens sont également naturellement utilisés dans de nombreux agents pour répondre à diverses problématiques où l'apprentissage entre en jeu.

Cette notion est très présente dans la littérature mais n'est pas partagée par toute la communauté. Wooldridge (Wooldridge, 2001) note d'ailleurs que dans certains cas, la capacité d'apprentissage de l'agent est à éviter dans des systèmes sensibles²² dont le comportement ne doit surtout pas changer au cours du temps. Peut-on pour autant considérer que ce type de système puisse encore être caractérisé comme système multi-agents ? En effet, si on pousse jusqu'au bout la notion d'autonomie, caractéristique essentielle de l'agent, on peut légitimement penser

²²Dans un système de contrôle du trafic aérien par exemple, il n'est pas du tout souhaitable que le système fasse librement évoluer son comportement.

que la capacité à changer de comportement au cours du temps en est une composante essentielle. Alors que l'autonomie d'un agent avait jusque là été définie par rapport à son environnement et aux instructions qu'il pouvait en recevoir, on parle ici d'une autonomie de l'agent par rapport à sa conception originelle. Cette capacité donne en effet la possibilité à l'agent de faire évoluer ses règles comportementales originelles, et ainsi d'échapper à une forme de déterminisme, et donc d'embrasser par cette émancipation plus pleinement encore cette notion d'autonomie. On peut néanmoins considérer que cette vision est une position extrême, imposant des contraintes lourdes et parfois non souhaitables pour le développement de l'agent. On voit encore une fois la difficulté d'aboutir à un consensus sur une définition commune de l'agent.

2.1.3 L'environnement

Tout agent est créé dans le but d'opérer dans un environnement. L'environnement comprend tout ce qui est extérieur à l'agent, que ce soit les objets, les caractéristiques du monde où il évolue ou les autres agents du système. Les environnements possibles dans lesquels sont amenés à évoluer les agents sont indénombrables que ce soit dans le monde physique ou des mondes virtuels. Pour n'en citer que quelques-uns, on peut donner comme exemples : la pièce qu'un agent aspirateur va devoir nettoyer, l'ensemble du réseau aérien qu'un système multi-agents va devoir contrôler, le plateau d'un jeu de Go où un système multi-agents va devoir chercher la meilleure stratégie, le sol martien qu'un robot conçu sur un SMA va explorer ou encore un univers virtuel où des agents avatars humains vont devoir cohabiter. Comme on le voit, ces environnements sont très diversifiés et leurs caractéristiques peuvent être très différentes. On réalise rapidement que, pour chaque environnement, un agent aura ses spécificités et qu'il est (aujourd'hui au moins) impossible de concevoir un agent universel pouvant s'intégrer à n'importe quel environnement. Russel et Norvig (Russell and Norvig, 1995) ont dégagé certains critères permettant de classer les environnements et ainsi de concevoir son agent en fonction des exigences de l'environnement.

Accessible/Inaccessible

Dans un environnement accessible, un agent peut obtenir des informations complètes, précises et à jour sur l'état de l'environnement à tout moment. On parlera donc d'accessibilité au niveau perceptuel pour l'agent. Un jeu d'échec peut être considéré comme un environnement accessible, mais la plupart des environnements sont en général inaccessibles. C'est le cas du monde physique tel que nous le connaissons et d'Internet par exemple. L'inaccessibilité peut être causée par plusieurs facteurs : contraintes physiques (limite de la perception visuelle par exemple), trop grande quantité d'information à analyser (c'est le cas d'Internet), etc. . L'accessibilité d'un environnement est une notion importante puisqu'il est bien évident qu'un agent pouvant accéder à la totalité des informations de son environnement (agent omniscient) aura la tâche plus aisée. Il est plus facile de prendre la bonne décision quand on maîtrise tous les paramètres. L'accessibilité au niveau des actions, c'est à dire la sphère qui limite les possibilités d'action d'un agent, n'est pas incluse dans ce concept selon Russel et Norvig. Néanmoins,

on peut également considérer que la plupart des environnements ne permettent pas à un agent d'agir sur tout (agent omnipotent).

Déterministe/Stochastique

Dans un environnement déterministe, les conséquences d'une action donnée sont connues, il n'y a pas d'incertitude quant au résultat de l'action (c'est à dire le nouvel état de l'environnement après que l'action ait été effectuée). Il est évidemment plus facile de concevoir des agents évoluant dans un environnement déterministe, leurs chances de succès étant plus élevées. Malheureusement, comme le soulignent Russel et Norvig, tout environnement quelque peu complexe est stochastique. Si plusieurs agents opèrent simultanément sur l'environnement, il est évident que les actions de l'un peuvent interférer sur le résultat attendu par un autre. Le non-déterminisme représente le fait qu'un agent n'ait qu'une sphère d'influence limitée dans son environnement (Wooldridge, 2001).

Épisodique/Non-épisodique

Un épisode consiste en une phase de perception suivie d'une seule action. Dans un environnement épisodique, l'expérience de l'agent peut être divisées en « épisodes ». Chaque épisode est alors indépendant des autres, le résultat de l'exécution d'une action ne dépend que de l'épisode courant. Dans un environnement séquentiel (ou non-épisodique), l'exécution d'une action peut influencer toutes les actions à venir. Cette caractéristique peut être graduée, c'est à dire que le périmètre d'influence de l'exécution d'une action peut être réduit à un nombre limité d'épisodes (un plan d'actions). Dans la pratique, un environnement épisodique est plus facile à gérer au niveau de la conception de l'agent car la combinatoire des possibilités conséquentes à une action est beaucoup moins vaste. Malheureusement, peu d'environnements peuvent prétendre à cette caractéristique.

Statique/Dynamique

Un environnement statique est un environnement qui ne peut subir de changements autres que ceux dus à l'intervention d'un agent. À l'inverse, un environnement dynamique subit des changements d'états imputables à d'autres processus. Ces changements échappent au contrôle de l'agent. De fait, les environnements dynamiques sont en général non déterministes, d'autres processus peuvent interférer avec la tâche qu'effectue un agent. Une fois encore, la plupart des environnements complexes peuvent être considérés comme dynamiques. Plus un environnement est dynamique, plus ses changements d'états sont fréquents et nombreux. Un environnement dynamique demande donc la conception d'agents particulièrement réactifs. Nous verrons plus en avant ce critère de réactivité d'un agent dans la partie 2.2 consacrée aux caractéristiques des différents types d'agent.

Discret/Continu

Un environnement peut être considéré comme discret s'il n'a qu'un nombre fini d'actions et de messages perceptifs. On peut de nouveau citer l'exemple du jeu d'échec en tant qu'environnement discret : le nombre de coups (les actions) est fini, et les dispositions des pions (messages perceptifs) également. Mais la plupart des environnements sont continus. Au niveau informatique, de nombreux environnements sont rendus discrets par l'architecture discrète du système basée sur le codage digital, et les variables continues sont en général approximées par un ensemble de valeurs discrètes. Donc, un agent ne peut percevoir les environnements à priori que sous forme discrète, même s'il peut accéder à une très bonne approximation d'un environnement continu par un système discret. En général, on pourra considérer comme continu tout environnement dont le nombre infini ou trop important d'états doit contraindre à une approximation discrète.

2.2 Les modèles d'agents

2.2.1 La problématique de la modélisation agent

Si le domaine des systèmes multi-agents peut aujourd'hui être considéré comme un domaine de l'informatique relativement mûr aux vues de ses 25 années d'existence, il a souffert de difficultés chroniques dans l'élaboration de modèles viables et largement réutilisables d'agents. De la même façon qu'il est difficile de dégager une définition consensuelle et commune d'un agent, on ne peut facilement le modéliser du fait de la diversité des applications pratiques des SMA qui ne permet logiquement pas l'élaboration d'un modèle unique. Néanmoins, diffuser et viabiliser le concept d'agent nécessite le développement de modèles éprouvés et réutilisables tout comme cela avait pu être fait pour le modèle objet. C'est dans les années 90 que de nombreux chercheurs ont défriché ce terrain en proposant plusieurs modèles. Cet effort de modélisation n'est pas terminé et de nombreuses recherches portent encore sur la création de nouvelles architectures d'agent permettant de répondre aux nouveaux domaines applicatifs auxquels les SMA s'ouvrent régulièrement comme la modélisation d'agents émotionnels. Néanmoins, ces travaux ont jeté les bases sur lesquelles de nombreux modèles actuels, dont le notre, et ont également permis d'envisager une typologie des architectures d'agent.

L'ensemble des architectures s'accordent sur le fait qu'un agent reçoive (ou perçoive) des flux (ou stimuli) de son environnement via des capteurs²³ (ou organes sensitifs), procède un traitement sur ces données et restitue une action sur l'environnement via ses effecteurs (ou organes moteurs). Ces actions sont elles-mêmes perçues par l'agent, ceci formant une boucle : on décrit l'agent comme étant *intégré* à son environnement (*embedded* selon (Ferguson, 1992)). Les architectures sont nombreuses mais chacune a pour objectif de doter l'agent d'un certain niveau d'autonomie et d'intelligence dans la phase du traitement des informations, et par conséquent dans l'acuité de ces décisions matérialisées par ses actions sur l'environnement.

²³sensor en Anglais.

	Agents cognitifs	Agents réactifs
Comportement téléonomique	Agents intentionnels	Agents pulsionnels
Comportement réflexe	Agents "modules"	Agents tropiques

TAB. 2.1 – Taxonomie des agents.

Plusieurs taxonomie existent. Franklin and Graesser (1997) en propose une adaptée et complétée par Drieu (2001) qui distingue ainsi : les agents réactifs, intentionnels, autonomes, adaptatifs, mobiles, flexibles et sociaux. Wooldridge (2001) distingue quant à lui deux grandes catégories : les agents réactifs et les agents cognitifs, en précisant également que des architectures hybrides existent organisées en couches horizontales ou verticales. Ferguson (1992) propose une taxonomie assez proche où il distingue les architectures délibératives, non-délibératives et hybrides. Deux critères orthogonaux semblent émerger de l'ensemble de ces taxonomies et permettre la distinction entre différents types d'agent : la type de représentation interne du monde et la base de la dynamique comportementale. Un agent possédant une représentation symbolique du monde qu'il utilise pour choisir ses actions est un agent cognitif, à la différence d'un agent réactif qui ne possède que des connaissances perceptuelles sur monde. Un agent animé par des buts est un agent téléonomique alors qu'un agent fournissant uniquement une réponse programmé à un stimulus donné est un agent réflexe. Cette catégorisation nous fourni le tableau 2.2.1. Aucune de ces taxonomies ne fait mention des architectures plus récentes d'agents émotionnels, souvent construit comme une surcouche aux agents délibératifs influant sur l'aspect comportemental.

En simplifiant, on peut effectuer une catégorisation de ces différentes architectures sur le mécanisme qui permet à l'agent de déterminer l'action qu'il va faire à partir de ses informations entrantes. Deux grands modes de raisonnement peuvent être définis, donnant ensuite une gamme de variations de l'un à l'autre. Dans le premier, les choix d'un agent procèdent d'une délibération sur les différentes options qui lui sont offertes (par exemple, via l'utilisation d'un modèle lexical interne, via une recherche dans ses plans d'actions, ou en considérant l'utilité prévisionnelle de chacune de ces options d'action). On parle donc d'agent délibératif. Dans le second, les choix de l'agent sont contextuels, c'est à dire que le contexte environnemental dans lequel il se situe va pleinement déterminer l'action qu'il va exécuter. Le choix est pré-programmé ou fixé pour s'exécuter sous un ensemble défini de conditions environnementales. On parle d'agent non-délibératif²⁴ ou réactif. Outre ces deux grands modèles, on trouve 2 autres types d'architecture intéressante. La première se situe à mi chemin du modèle cognitif et du modèle réactif, utilisant des combinaisons de techniques délibératives et non délibératives pour le choix des actions. On parle alors d'architecture hybride. Enfin, et c'est ceux qui nous intéressent le plus, on trouve des modèles où le but est de modérer par un ressenti émotionnel la rationalité de l'agent dans son processus délibératif. On parle dans ce cas de modèle d'agent émotionnel.

²⁴La distinction entre agents délibératifs et non délibératifs est comparable à la différence faite par Simon (Simon, 1981) entre les systèmes manifestant une rationalité procédurale, c'est à dire les système qui calculent l'acte rationnel à effectuer, et ceux manifestant une rationalité substantive qui exécutent simplement l'acte.

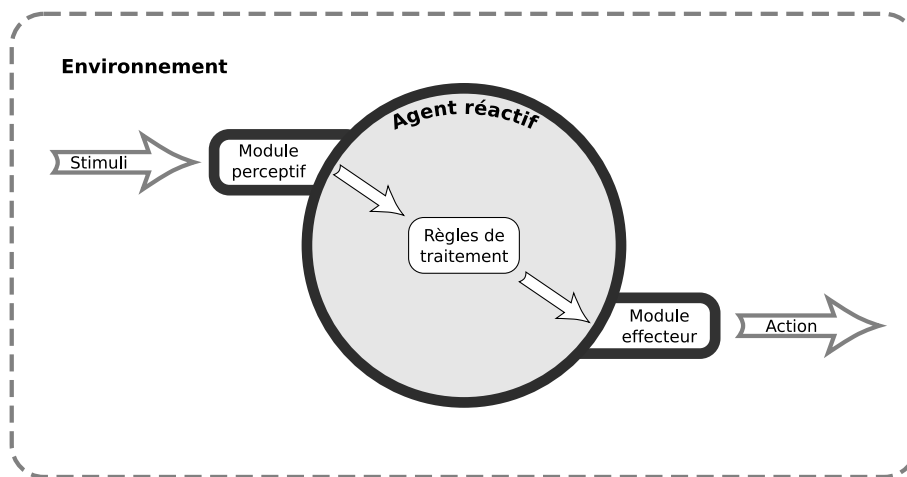


FIG. 2.1 – Agent réactif.

2.2.2 L'agent non-délibératif

L'agent non-délibératif puise ses principes de fonctionnement dans les préceptes de l'IA comportementale selon lesquels on peut assister à l'émergence de comportements intelligents chez des entités ayant pourtant des principes de fonctionnement extrêmement simplifiés. Un comportement complexe n'est pas forcément le fait d'un modèle complexe, mais plutôt d'un environnement complexe (Simon, 1981). On attribue généralement les premiers agents de ce type - également nommés agents réactifs, situés²⁵, ou comportementaux - à Brooks (Brooks, 1986). Dans ce type d'architecture, l'agent effectue sa prise de décision intégralement à l'exécution à partir d'un ensemble limité d'informations (en général, uniquement celles perceptibles à cet instant par leurs capteurs), d'un état interne limité et avec une part d'inférence limitée (Hanks and Firby, 1990)²⁶. Si d'une architecture non-délibérative à l'autre, l'usage d'un état interne peut varier, elles ont toutes en commun l'absence d'une connaissance complète et juste de leur environnement à tout instant.

Là où des systèmes de planification classiques, utilisés dans beaucoup d'architectures délibératives, sont conçus pour produire une solution optimale sinon correcte, ces architectures délivrent des solutions robustes. Pour se faire, ces modèles s'appuient sur des mécanismes simples tels que les automates à états finis ou des règles de stimulus-réponse spécifiques au domaine. Ce type d'agent, par sa simplicité, démontre évidemment des performances très intéressantes en terme de temps de calcul et de robustesse, qui sont particulièrement appréciées en robotique et dans les systèmes temps réels. L'agent réactif est composé d'un module perceptif (ou senso-

²⁵Cette dénomination, initialement *situated* en anglais, provient du fait que ce type d'agent exerce sa décision uniquement à partir des informations contextuelles dont il dispose à un moment donné. C'est donc sa situation dans l'environnement qui définit complètement sa réaction

²⁶Dans le cas extrême où l'agent n'a pas d'état interne (son activité est donc totalement déterminée par son environnement), on parle d'agent *tropique* (Genesereth and Nilsson, 1987) en référence au mécanisme d'orientation des organes d'une plante selon l'anisotropie du milieu.

riel) la plupart du temps directement relié à un module moteur (ou effecteur) comme l'illustre la figure 2.1. Dans la modélisation d'un agent réactif, on fait correspondre à un stimulus perçu une action à exécuter par l'agent. On retrouve ce principe dans la nature, qui a beaucoup inspiré l'IA comportementale, notamment au niveau des fourmis qui sont un exemple typique de l'intelligence collective qui peut émerger de l'interaction d'entités au comportement très simple. Ces agents sont d'ailleurs modélisés pour simuler le comportement de certaines colonies d'insectes. Bien que très simplifiés, les agents réactifs présentent des caractéristiques qu'il peut être intéressant d'utiliser dans le cadre de notre simulation.

Si on se réfère à la définition formelle de l'agent donnée précédemment, celle-ci peut généralement être fortement simplifiée pour les agents non délibératifs :

Agent $N = (\mathcal{P}, \mathcal{A}, f)$ où :

- $\mathcal{P} = \{p_1, p_2, \dots, p_i\}$: l'ensemble des perceptions de N ,
- $\mathcal{A} = \{a_1, a_2, \dots, a_k\}$: l'ensemble des actions réalisables par N ,
- $f : \mathcal{P}^{l^*} \rightarrow \mathcal{A}$: la fonction de choix de N .

Dans cette nouvelle définition, on constate la disparition de l'ensemble des connaissances de l'agent ainsi que de la fonction des possibles et de celle de l'utilité. En effet, dans la plupart de agents non délibératifs, aucune connaissance n'est stocké par l'agent, et un ensemble déterminé de stimuli en entrée détermine complètement l'action de l'agent. Une et une seule action est donc associée à un ensemble de stimuli, la fonction des possibles n'a donc plus de raisons d'exister, et a fortiori celle d'utilité puisqu'il n'est plus nécessaire de déterminer l'action la mieux adaptées. Cette définition est néanmoins extrêmement simplificatrice et on verra dans les architectures suivantes qu'ils existent des agents maintenant une connaissance minimum et ayant un processus délibératif embryonnaire. Il existe en réalité tout une gamme de possibilités entre la non délibération et la délibération.

L'architecture à subsomption - Brooks

Cette architecture a été développée par Brooks pour contrôler des robots. Ce modèle, à la fois architecture et méthodologie de conception, est basé sur une organisation en couches de l'agent. A la différence des architectures en couches classiques à décomposition horizontale où chaque couche assure une fonction de contrôle (perception, mémoire, attention, ...), chacune des couches est ici spécialisée pour traiter un problème spécifique : évitement d'obstacle, exploration de l'espace, ... ; on parle de décomposition verticale. Chaque couche fonctionne sur la base d'un automate à états finis pouvant changer d'état à l'arrivée d'un message d'autres couches ou après l'écoulement d'un laps de temps. Les couches opèrent en parallèle de manière asynchrone. Les couches sont organisées en niveaux où les niveaux les plus hauts peuvent prendre le contrôle sur les niveaux inférieurs, notamment en substituant les entrées ou en supprimant les sorties afin de modifier le flux normal des informations. Une bonne organisation de ces niveaux conduit l'agent à satisfaire les « buts » des niveaux les plus hauts. Dans l'exemple d'un robot, on peut par exemple préférer que l'évitement d'obstacle est une priorité plus haut que l'exploration afin que le robot ne reste pas bloqué en butée contre un obstacle.

2.2.3 L'agent délibératif

L'agent cognitif hérite en partie des principes de l'IA classique. Il possède en effet un centre cognitif (ou délibératif) qui analyse les stimuli perçus pour déterminer le comportement optimal à adopter. Ce type d'agent s'inspire du fonctionnement humain qui traite certaines informations avant d'agir. Les principes utilisés pour construire ce type d'agent sont assez complexes et nombreux. Une structure s'est néanmoins dégagée comme un standard : le modèle BDI²⁷, basée sur les travaux de Bratman (Bratman, 1987; Bratman et al., 1988) et formalisée d'abord par Cohen and Lesveque (1990), puis par Rao et Georgeff (Rao and Georgeff, 1991, 1995)²⁸. Nous allons présenter quelques notions de base des agents BDI ainsi que quelques-unes des structures BDI existantes.

Beliefs, Desires et Intentions

Bratman est le premier philosophe à avoir dégagé l'importance des intentions dans le raisonnement. Selon lui, les intentions consistent en un plan d'action devant être compatible avec les croyances et les buts de l'individu, alors qu'auparavant, l'intention était conçue comme étant le produit des croyances et des buts. L'intérêt est que cela permet de prendre en compte les ressources limitées en terme de temps dont dispose un agent pour choisir une action à exécuter afin d'atteindre ses buts avant que l'environnement soit passé dans un état où le choix sera devenu obsolète. Le raisonnement de l'agent est ainsi rendu moins exhaustif mais plus pertinent face à la dynamique de l'environnement. La délibération de l'agent prend place dans un ensemble de croyances qu'il maintient et met à jour et par rapport à ses désirs qu'il peut également remettre en cause. Les intentions doivent être :

- révisables, pour pouvoir être remise en question si elles deviennent obsolètes (Bratman et al., 1988),
- robustes, pour se maintenir suffisamment malgré l'évolution permanente de l'environnement (Bratman et al., 1988),
- cohérentes, pour qu'un agent ne poursuive pas des buts contradictoires (Bratman et al., 1988),
- réalisable, car un agent doit croire qu'il est capable de réaliser son intention (Cohen and Lesveque, 1990),
- réaliste²⁹, car un agent doit croire que son intention est possible (Cohen and Lesveque, 1990).

Cohen et Levesque ont introduit le concept de mondes possibles (passé, présent et futur) pour l'agent, ainsi que des gradations dans les intentions de l'agent qui peuvent être fanatiques (maintenues à tout prix) ou raisonnées (remises en cause). Rao et Georgeff ont quant à eux étendu le

²⁷Pour *Belief, Desire, Intention* (Croyance, Désir, Intention en Français). Le terme *désir* est à prendre dans le sens d'envie, objectif ou but.

²⁸Konolige and Pollack (1993) ont également proposé leur formalisation qui s'avère plus complexe et ne sera pas présentée ici.

²⁹Le fait qu'une intention soit réalisable est relatif à l'agent, alors que celui qu'elle soit réaliste est relatif à l'environnement.

concept de monde possible en monde accessibles en croyance, en but et en intention, respectivement l'ensemble des situations que l'agent croit être possibles, désirerait atteindre et va tenter d'atteindre. Pour chacun de ces auteurs, on trouve une logique modale permettant l'expression de ces concepts.

Délibération, planification, éveil

La structure BDI induit d'autres notions. La planification est ainsi le processus qui consiste pour un agent à mettre à partir de ses croyances et ses buts à réviser ses intentions afin de choisir les actions qu'il va accomplir. La différence entre la planification pour un agent BDI et celles développées antérieurement en IA est qu'elle ne souffre pas du principe d'exhaustivité selon lequel pour faire le meilleur choix il était nécessaire de posséder l'ensemble des possibilités d'action et leurs conséquences. La délibération est un processus général comprenant la planification et la mise à jour des croyances et des buts en fonction de l'évolution de l'environnement. L'éveil est la capacité d'un agent à percevoir les modifications de son environnement assez rapidement afin que son processus de délibération ne produise pas de résultats obsolètes. Le modèle BDI, bien que bien défini, n'est pas simple à mettre en œuvre. Le mécanisme de planification doit produire rapidement des résultats pertinents. Le plus souvent, on intègre des bibliothèques de plans à l'agent qui définissent les tâches qu'il peut exécuter et des plans face à des situations fréquentes. Cela correspond en fait au fonctionnement humain où un certain nombre d'actions quotidiennes sont rendues machinales. L'éveil de l'agent est également dépendant de ses performances sensibles mais également du degré de dynamisme de l'environnement. Un agent qui sait tirer partie de ses expériences en stockant les plans qu'il a élaboré lui-même est plus autonome et plus éveillé car il est capable de répondre plus rapidement aux situations qu'il a déjà rencontrées.

Structures BDI

La figure 2.2 présente une architecture classique d'agent BDI avec d'une part le processus de délibération, et d'autre part les processus de révision en fonction des modifications de l'environnement pour les croyances, buts et intentions et en fonction des nouveaux plans générés pour la bibliothèque de plans.

Il existe plusieurs architectures existantes d'agents BDI. D'abord, Bratman et al. (1988) a proposé sa propre architecture BDI baptisée IRMA³⁰. L'agent IRMA possède 4 structures permettant de stocker les croyances, les buts, les intentions et les plans, un analyseur moyens-fin pour choisir le plan approprié, un analyseur d'opportunités pour déceler les opportunités qui se présentent dans l'environnement, un moteur d'inférence pour raisonner sur l'environnement et enfin un filtre et un module de délibération pour sélectionner les intentions. Ensuite, l'architecture PRS³¹ de Georgeff and Lansky (1987) est une des plus répandues. On rencontre bien évidemment des similarités avec l'agent IRMA notamment pour les 4 structures de données.

³⁰Pour *Intelligent Resource-Bounded Machine Architecture*.

³¹Pour *Procedural Reasoning System*.

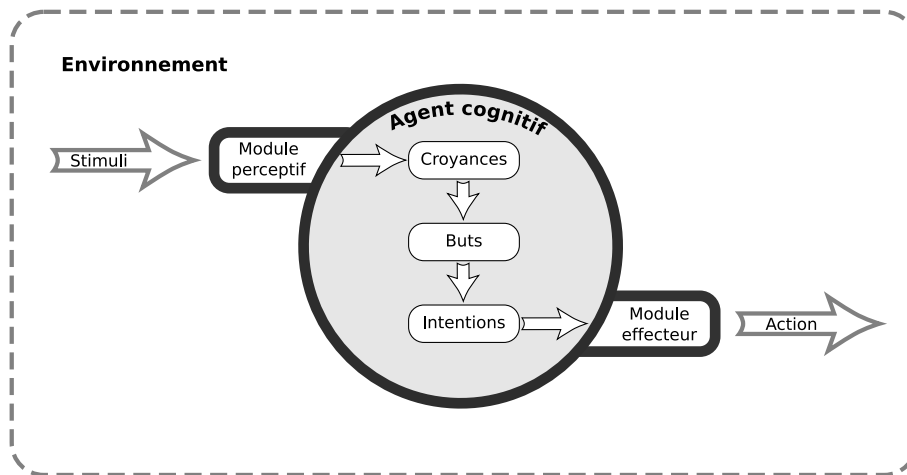


FIG. 2.2 – Agent BDI.

Néanmoins, les types de stockage sont différents : une base de donnée pour les croyances et l'observation de l'environnement, une structure sous forme d'états valides dans des intervalles de temps pour les buts, une bibliothèque de plans avec des conditions de déclenchement et une structure stockant les intentions sous forme de plans choisis pour être exécutés. Un moteur de raisonnement central assume la partie fonctionnelle de l'agent. Citons encore Wooldridge et al. (1997) qui a proposé une adaptation de PRS baptisée dMARS.

2.2.4 Les agents hybrides

Les agents structurés en couches ou agents hybrides correspondent à des modèles qui mélangent les différentes architectures que nous venons de voir. On les appelle *agents structurés en couches* parce qu'en général l'intégration des différentes fonctionnalités propres à chaque architecture est réalisée par un partitionnement en couches de la structure de contrôle de l'agent, chaque couche correspondant le plus souvent à une caractéristique. Les agents émotionnels que nous allons voir présentent pour certains (Camurri and Coglio, 1998; Bisognin and Pesty, 2004) cette structure qui vient greffer par dessus un agent BDI une couche émotionnelle. D'autres structures visent à reproduire le mode de fonctionnement humain où l'on distingue les réactions réflexes, les tâches routinières et les actions demandant une réflexion. Les actions réflexes sont alors prises en charge par une couche correspondant à un agent réactif et les autres par des structures d'agents cognitifs travaillant à des niveaux plus ou moins abstraits³². Par rapport aux flux d'entrée et de sortie, ces couches peuvent être organisées horizontalement, verticalement avec un seul passage ou verticalement avec deux passages comme l'illustre la figure 2.3. Ces types d'architecture sont utilisés par Camurri and Coglio (1998) (en couches horizontales) et Bisognin and Pesty (2004) (en couches verticales).

³²Les actions réfléchies et construites par l'agent nécessitent en effet d'élaborer des métamodèles d'actions, de plan, de langage de communication sur lesquels un moteur cognitif peut ensuite être conçu afin de fournir des raisonnements plus haut niveau.

gnin and Pesty (2004) (en couches horizontales avec un sous module plus complexe) pour leurs agents émotionnels. On la trouve également dans l'architecture *TouringMachines* (Ferguson, 1992) organisée horizontalement et celle de *InteRRaP* organisée verticalement à deux passages. L'intérêt de ces architectures hybrides est qu'elles permettent de faire cohabiter dans un agent des aspects réactifs, cognitifs et émotionnels, et de modéliser plus fidèlement le comportement humain.

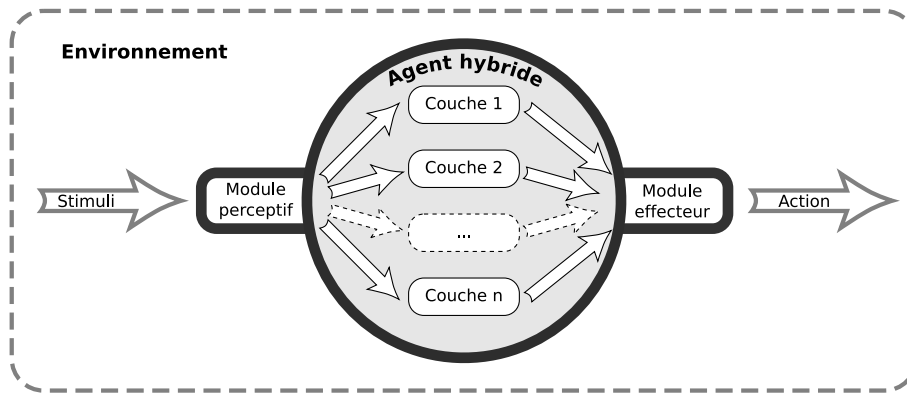
2.2.5 L'agent émotionnel

Il y a relativement peu de temps que l'on s'intéresse vraiment à la possibilité de pouvoir obtenir de la part d'un ordinateur autre chose que la manifestation d'une forme d'intelligence humaine. Même s'il est reconnu depuis longtemps que les émotions jouent un rôle primordial dans le processus décisionnel humain, la modélisation des émotions en informatique n'a pas connu le même essor que celle des comportements intelligents. Néanmoins, dans le contexte de la simulation de comportements humain notamment, on s'aperçoit que sans les émotions on perd une part de fidélité par rapport au modèle réel. Les émotions sont à l'origine de comportements non rationnels qui ne sont pas pris en compte par les deux modèles d'agents que nous venons de voir. On attribue à Simon (1967) en 1960 les premières considérations pour les émotions en IA, puis réellement en 1980 avec Sloman (2001) les débuts de la recherche sur le sujet. Les architectures d'agents émotionnels sont quant à elles beaucoup plus récentes et datent du milieu des années 90.

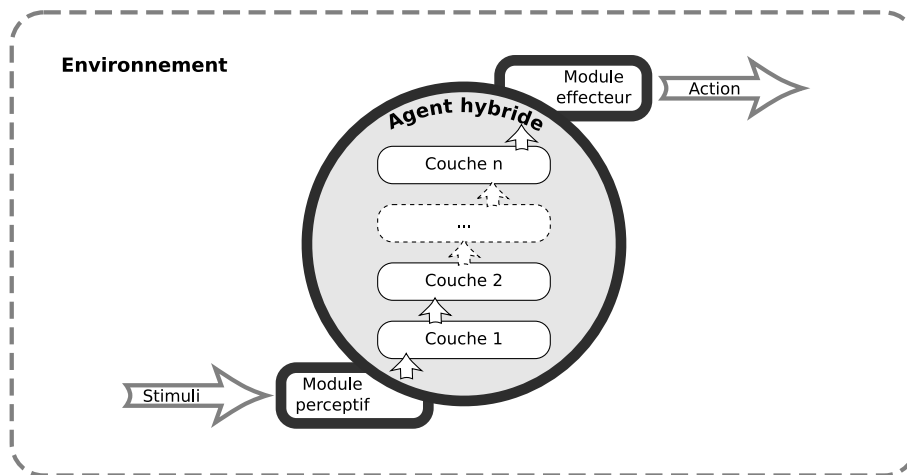
Nous n'entrerons pas dans le détail de ces structures. Simplement, le principe est de considérer un module supplémentaire qui est intégré à l'agent. Suivant les architectures, l'importance, le rôle et la construction de ce module peuvent varier. En général, il se présente néanmoins comme une partie de l'agent qui oriente son comportement pour qu'il corresponde à son état émotionnel. Celle de Velásquez (1997) se situe par exemple plus en sortie sur la partie moteur de l'agent. Celle de Camurri and Coglio (1998) et celle de El-Nasr and Skubic (1998) s'intègrent complètement dans le processus de prise de décision de l'agent et oriente ses choix. Enfin, celle de Bisognin and Pesty (2004) opère au niveau de la communication de l'agent. On ne trouve pas encore d'architecture prenant en compte l'ensemble des rôles que peuvent jouer les émotions dans le comportement d'un humain.

2.3 Les interactions entre agents

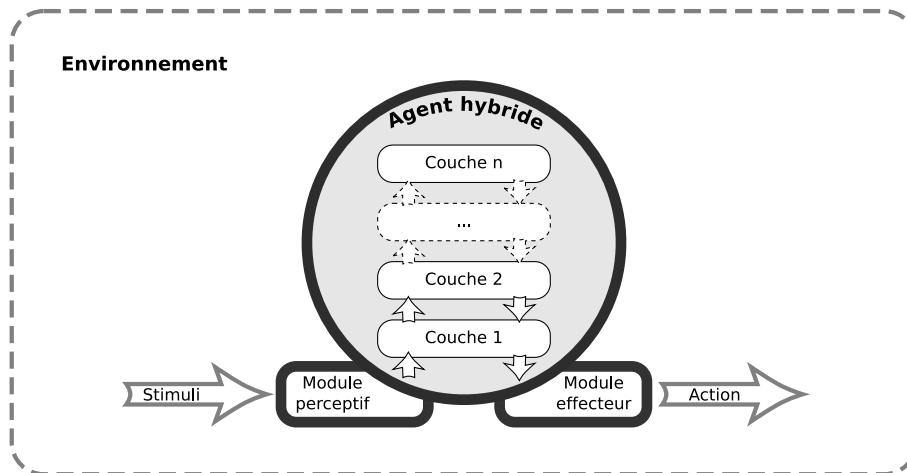
Les chapitres précédents ont abordé les notions d'agents et d'environnement. Le descriptif de ce qu'est un agent, ou plutôt un système multi-agents, ne saurait être complet sans l'aspect fondamental des interactions qu'un agent peut entretenir avec d'autres entités de son environnement qu'il s'agisse d'agents ou de simples objets. L'agent en lui-même n'est ainsi pas suffisant. Ce sont les interactions qu'il va avoir avec son environnement et surtout avec les autres agents qui possèdent un intérêt. Dans cette partie, nous allons donc présenter le principe de communication entre agents par envoi de messages, les langages aujourd'hui standardisés permettant



(a) Horizontales



(b) Verticales à un passage



(c) Verticales à deux passages

FIG. 2.3 – Agents hybrides (architectures en couches).

de mettre en œuvre ce type de communication ainsi que les procédures de communication permettant l'existence d'un comportement social. Ce processus de communication est essentiel, surtout dans le contexte de la simulation comportementale. Il ne peut néanmoins avoir d'intérêt que si l'agent possède déjà une structure lui permettant de gérer ces stimuli. Un slogan populaire de la communauté multi-agents est : « *There's no such thing as a single agent system.* »³³. Du point de vue de la systémique, cela signifie que tout système est une partie d'un plus grand tout. Cela nécessite donc une bonne modélisation d'une part de l'agent, et d'autre part de ses interactions.

2.3.1 Les interactions entre agents

Ferber (1995) propose la définition suivante de l'interaction :

Une interaction est une mise en relation dynamique de deux ou plusieurs agents par le biais d'un ensemble d'actions réciproques.

(Ferber, 1995)

Les agents interagissent à travers un ensemble d'événements pendant lesquels les agents sont en relation les uns avec les autres, soit directement soit par le biais de l'environnement. Toujours d'après J. Ferber, la notion d'interaction suppose :

- La présence d'agents capables d'agir et/ou de communiquer.
- Des situations susceptibles de servir de point de rencontre entre agents : collaboration, déplacement de véhicules amenant à une collision, utilisation de ressources limitées, régulation de la cohésion d'un groupe.
- Des éléments dynamiques permettant des relations locales et temporaires entre agents : communication, choc, champ attractif ou répulsif, etc.
- Un certain « jeu » dans les relations entre les agents leur permettant à la fois d'être en relation, mais aussi de pouvoir se séparer de cette relation, c'est-à-dire de disposer d'une certaine autonomie. Si des agents sont totalement liés par un couplage fixe, leur interaction devient rigide et ils n'interagissent plus au sens plein du terme.

(Ferber, 1995)

L'intelligence individuelle d'un agent émerge en fait de l'aspect pluriel des Systèmes Multi Agents. L'intelligence ne peut ainsi être comprise d'un point de vue individuel, c'est le résultat d'un effort commun d'échanges, de partages et d'interdépendances d'un ensemble d'agents. En l'absence d'interactions, un agent devient isolé et relativement figé dépourvu de situation où il lui est nécessaire de s'adapter. Les circonstances dans lesquelles des agents entrent dans un processus interactif définissent et donnent des critères de comparaison entre les différents types d'interaction. Ces circonstances permettant l'interaction entre les agents sont appelées *situations d'interaction* dont voici la définition :

³³Traduisible par « *Il n'y a rien de tel qu'un système mono-agent.* » avec le double sens que les systèmes mono-agent n'existent pas, et que pourtant le principe agent est le meilleur qu'on puisse trouver.

On appellera situation d'interaction un ensemble de comportements résultant du regroupement d'agents qui doivent agir pour satisfaire leurs objectifs en tenant compte des contraintes provenant des ressources plus ou moins limitées dont ils disposent et de leurs compétences individuelles.

(Ferber, 1995)

2.3.2 La communication entre agents

La notion de communication

La communication entre agents est devenue un axe de développement très étudié par la communauté multi-agents. Le développement d'Internet et des SMA destinés à ce domaine y est pour beaucoup car il nécessite la définition de standards permettant de faire collaborer des agents hétérogènes. La communication entre agents est plus complexe que celle entre objets du fait que les agents sont autonomes et ne partagent pas forcément les mêmes objectifs (Wooldrige, 2001). Dans un contexte objet, la communication consiste en un appel de méthode que l'objet concerné exécute. Dans un contexte agent, l'invocation directe d'une méthode permettant de faire exécuter par l'agent une action n'est pas permise, la communication est donc un processus d'envoi de messages afin qu'ils soient traités par le destinataire. La communication intervient à plusieurs niveaux. Le plus simple consiste à échanger des informations dans le simple but d'enrichir mutuellement ses connaissances, au niveau supérieur on peut trouver des processus de communication permettant à une entité d'en solliciter une autre, puis des processus permettant la synchronisation des tâches effectuées par plusieurs entités en parallèle et enfin des processus permettant à un ensemble d'entité de collaborer pour la réalisation d'une ou plusieurs tâches.

La théorie des actes de parole³⁴

Austin et Searle ont chacun développé une théorie considérant la communication comme une action. Selon Austin (1962), certains types d'expression du langage³⁵ sont comparables à des actions en ce sens qu'elles modifient l'état du monde extérieur (l'environnement pour l'agent). Austin donne l'exemple de la bénédiction du mariage « *Vous êtes maintenant mari et femme.* ». Austin distingue des verbes de demande, d'information et de promesse, et des actes de *locution* (« *Peux-tu m'aider à porter le paquet ?* »), d'*illocution* (« *Il m'a demandé de l'aider.* ») et ceux de *perlocution*³⁶ (« *Il m'a amené à l'aider.* »). Searle (1969) a étendu la théorie des speech acts en y ajoutant les conditions dans lesquelles peuvent se dérouler les échanges de message :

- le destinataire du message doit être capable de le recevoir,
- l'émetteur doit être dans des conditions qui lui permettent de formuler son message, s'il

³⁴Speech act theory, en Anglais.

³⁵Natural language utterances, en Anglais.

³⁶Locutionary, illocutionary and perlocutionary acts, en Anglais.

s'agit d'une illocution le destinataire doit être capable d'effectuer l'action demandée et l'émetteur doit croire que le destinataire en est capable,

- l'émetteur doit être sincère et ne pas formuler des requêtes d'illocution sollicitant l'accomplissement d'une action qu'il ne désire pas.

Il a également dégagé une classification des actes de parole plus fine que celle d'Austin :

- message représentatif, contenant une information sur l'environnement,
- message directif, contenant une demande,
- message promissif, contenant une promesse,
- message expressif, expression d'un état interne,
- message déclaratif, entraînant un changement d'état institutionnel.

La planification des actes de parole

Cette théorie a entraîné l'intégration, par les chercheurs en AI, de l'acte de parole dans les processus de planification en tant qu'action permettant d'atteindre un objectif. Cohen and Perrault (1979) ont été les premiers à se pencher sur l'intégration de ces actes de parole dans le processus de planification. L'idée poursuivie par d'autres ultérieurement est d'appliquer le principe de pré-conditions et de post-conditions déjà utilisé pour les actions classiques aux actes de paroles. Cela permet de modéliser l'impact ou l'utilité que pourra avoir tel ou tel message pour la réalisation d'un but.

2.3.3 La notion de concurrence

La notion de concurrence est inhérente à la notion d'agent. L'autonomie de l'agent lui permet de gérer seul ses objectifs. Dans un système SMA, cela induit qu'il peut y avoir autant d'objectifs différents et donc potentiellement concurrents qu'il y a d'agents. La notion de concurrence apparaît dès lors que les agents partagent des ressources³⁷. Lechilli et Chaib-Draa ont présenté une classification des relations entre agents qui synthétise et étend les travaux précédents sur le sujet. Cette classification fournit également une notation matricielle permettant par exemple de détecter les éventuels conflits. On peut ainsi distinguer :

- des relations entre les activités des agents qui peuvent travailler en commun,
- une différenciation des rôles des agents qui peuvent effectuer des tâches identiques ou différentes,
- des dépendances par rapport aux tâches si les tâches sont dépendantes ou indépendantes,
- des dépendances par rapport aux buts qui peuvent être concurrent ou non.

Ces notations fournissent un cadre formel aux processus de négociation et de collaboration que peuvent entretenir les agents entre eux. On peut ainsi savoir si ce type de relation est possible et comment la mettre en œuvre via un processus d'échange de messages.

³⁷Nous ne nous intéresserons pas ici au problème du partage des ressources informatiques également présent dans les SMA (plusieurs processus concurrents).

TAB. 2.2 – Performatives KQML.

Performative	Signification
<i>achieve</i>	S veut que R fasse réaliser quelque chose dans l'environnement.
<i>advise</i>	S déclare être prêt à recevoir une performative.
<i>ask-about</i>	S désire connaître toutes les phrases du vocabulaire de R.
<i>ask-all</i>	S désire connaître toutes les réponses de R à une question C.
<i>ask-if</i>	S désire savoir si la réponse à une question C est dans le vocabulaire de R.
<i>ask-one</i>	S désire connaître une des réponses de R à une question C.
<i>break</i>	S veut rompre un tube (cf. pipe).
<i>broadcast</i>	S veut envoyer une performative sur tous les ports.
<i>broker-all</i>	S désire recevoir toutes les réponses à une performative.
<i>broker-one</i>	S désire que R demande de l'aide afin de répondre à une performative.
<i>deny</i>	La performative concernée ne convient plus à S.
<i>delete-all</i>	S désire que R efface toutes les phrases correspondant à C de son vocabulaire.
<i>delete-one</i>	S désire que R efface une des phrases correspondant à C de son vocabulaire.
<i>discard</i>	S ne veut pas que R continue à répondre à une requête.
<i>eos</i>	Fin d'un flux de réponse à une requête.
<i>error</i>	S considère que le dernier message de R était mal construit.
<i>evaluate</i>	S désire que R évalue de façon simplifiée C.
<i>forward</i>	S désire que R fasse suivre un message à un autre agent.
<i>generator</i>	Identique à un standby pour un stream-all.
<i>insert</i>	S demande à R d'ajouter un contenu à son vocabulaire.
<i>monitor</i>	S désire les dernières réponses mises à jour de R à un stream-all.
<i>next</i>	S désire la prochaine réponse de R à une performative précédemment envoyée.
<i>pipe</i>	S désire faire suivre toutes les prochaines performatives à un autre agent.
<i>ready</i>	S est prêt à répondre à la dernière performative de R.
<i>recommend-all</i>	S désire connaître tous les noms des agents pouvant répondre à C.
<i>recommend-one</i>	S désire connaître le nom d'un agent pouvant répondre à C.
<i>recruit-all</i>	S désire que R récupère tous les agents pouvant répondre à C.
<i>recruit-one</i>	S désire que R récupère un agent pouvant répondre à C.
<i>register</i>	S peut envoyer des performatives à un agent donné.
<i>reply</i>	Communique une réponse attendue.
<i>rest</i>	S désire les réponses restantes de R à une performative précédemment envoyée.
<i>sorry</i>	S ne peut pas fournir une réponse plus complète.
<i>standby</i>	S désire que R se tienne prêt à répondre à une performative.
<i>stream-about</i>	Version en réponses multiples de ask-about.
<i>stream-all</i>	Version en réponses multiples de ask-all.
<i>subscribe</i>	S désire les dernières réponses mises à jour de R à une performative.
<i>tell</i>	S déclare à R que C est dans son vocabulaire.
<i>transport-address</i>	S associe un nom symbolique à une adresse de transport.
<i>unregister</i>	Correspond au deny d'un register.
<i>untell</i>	S déclare à R que C n'est pas dans son vocabulaire.

2.3.4 Les langages de communication entre agents³⁸

KQML

KQML est le premier langage standardisé ayant été adopté pour les processus de communication entre agents. Chaque message peut ainsi être caractérisé par une performative pouvant être considérée comme la classe du message et des paramètres pouvant être considérés comme les attributs de cette classe. Le tableau 2.2 présente les différentes performatives des messages KQML et le tableau 2.3 présente les différents paramètres d'un message KQML. Ce langage contient les bases nécessaires de tout langage de communication agent et les attributs permettant de définir le langage d'expression et l'ontologie du message lui assure une certaine extensibilité. Néanmoins, il souffre de quelques défauts : son manque de contraintes, l'absence de performative commit, l'absence d'une réelle définition formelle et l'absence d'une sémantique claire. C'est pour ces raisons parmi d'autres qu'a été créé le langage FIPA ACL.

³⁸Souvent appelés ACL pour *Agent Communication Language*.

TAB. 2.3 – Attributs des messages KQML.

Paramètre	Signification
:content	Contenu du message.
:force	Auquel cas l'expéditeur du message ne niera jamais son contenu.
:reply_with	Dans le cas où l'expéditeur attend une réponse (avec l'identifiant à donner à cette réponse).
:in-reply-to	Référence au paramètre reply-with.
:sender	Expéditeur.
:receiver	Destinataire.
:language	Langage dans lequel est écrit le contenu du message.
:ontology	Ontologie définissant la sémantique du message.

TAB. 2.4 – Performatives FIPA ACL (Wooldridge, 2001).

Performative	Envoi d'informa- tion	Demande d'informa- tion	Négociation	Réalisation d'ac- tions	Interception d'er- reur
accept-proposal			✓		
agree				✓	
cancel		✓			
cfp			✓		
confirm	✓				
disconfirm	✓				
failure					✓
inform	✓				
inform-if	✓				
inform-ref	✓				
not-understood					✓
propagate				✓	
propose			✓		
proxy				✓	
query-if		✓			
query-ref		✓			
refuse				✓	
reject-proposal			✓		
request				✓	
request-when				✓	
request-whenever				✓	
subscribe		✓			

FIPA

Ce langage est une initiative de la FIPA. Il s'inspire en partie de KQML en essayant d'en supprimer les défauts. À la place des 40 performatives de KQML, FIPA ACL en propose 22 décrites dans le tableau 2.4. Une des lacunes de KQML étant le manque de sémantique, les développeurs du langage FIPA ACL ont voulu doter leur langage d'une sémantique claire. Ils ont ainsi défini, en se basant sur les travaux de Cohen et Levesque sur les agents BDI, un langage formel baptisé SL permettant entre autres d'exprimer les croyances et les désirs des agents. Chaque message doit être conforme à une contrainte de faisabilité pour être conforme au format FIPA ACL. Enfin, SL définit les pré-conditions et les post-conditions d'un message. Aujourd'hui, le langage FIPA ACL est utilisé par la plupart des systèmes multi-agents développés.

Conclusion

Dans ce chapitre, nous avons présenté un large état de l'art des systèmes multi-agents abordant à la fois les aspects historiques, de l'agent en tant qu'entité individuelle et des interactions entre les entités d'un système. L'agent *BDI* que nous avons présenté dans la première partie s'avère particulièrement adapté à la modélisation du comportement humain puisqu'il est inspiré pour une bonne part de travaux sur la psychologie humaine. Il est de plus largement utilisé et éprouvé au sein de la communauté multi-agents en faisant une référence incontournable des modèles d'agent cognitif. Nous avons également présenté quelques références sur le plan des agents émotionnels montrant ainsi que le domaine était encore jeune et qu'aucun consensus n'émergeait pour favoriser un modèle plutôt qu'un autre. C'est donc logiquement que nous avons choisi le modèle *BDI* pour l'étendre afin de créer notre propre modèle d'agent émotionnel.

Pour ce qui concerne les interactions entre agents, il paraît clair que l'utilisation d'un langage de communication entre agents (*ACL*) est un choix incontournable puisque cela permet de s'appuyer sur un standard reconnu et construit de manière pertinente pour couvrir un large éventail de cas d'interaction entre deux agents ou plus. Le standard *ACL* de la *FIPA* est le plus récent, apportant un grand nombre d'amélioration par rapport à son prédécesseur : *KQML*, et également le plus largement utilisé. Le chapitre 4 détaillera notre utilisation de ces modèles à la fois au niveau interne de l'agent et de celui des interactions.

3

Fouille de données multidimensionnelles

Ce chapitre présente la spécificité des données issues de la simulation, notamment de par leur caractère multidimensionnel et temporel, par rapport aux données objets de la fouille de données d'ordinaire.

Sommaire

3.1	Historique, objectif et principe général de la fouille de données	68
3.2	La simulation et la problématique des données multidimensionnelles . .	70
3.2.1	Les dimensions spatiale et temporelle	70
3.3	Techniques et outils de stockage de données multidimensionnelles . . .	75
3.3.1	La notion de temps et les logiques temporelles	75
3.3.2	Les logiques temporelles	75
3.3.3	L'intégration du temps et de l'espace dans les SGBD relationnels et objets	76
3.4	Data mining temporel	79
3.4.1	Préliminaires	80
3.4.2	Les tâches classiques d'étude de données temporelles	82
3.4.3	ECD sur des données temporelles : la découverte de motifs	86
3.4.4	La méthode des projections aléatoires	91
3.5	Data mining spatial	93
3.5.1	Les techniques de data-mining spatial	93
3.5.2	Notion de sémantique d'une position et problématique	94

Introduction

Un outil de simulation n'a d'intérêt que si on peut garder la trace de l'évolution d'éléments intéressants pendant la simulation. Plusieurs problèmes se posent alors. Comment caractériser

les éléments intéressants ? Quelle structure de donnée va permettre de stocker les données de simulation ? Comment traiter les données obtenues de manière efficace pour qu'elles soient exploitables par l'expert, lui permettent de reconnaître des situations réelles caractéristiques et qu'il puisse même y trouver de nouvelles connaissances ? L'ensemble de ces problématiques rejoignent des questions fondatrices de la fouille de données (ou data mining (Fayyad et al., 1996)) et peuvent être synthétisées ainsi : comment exploiter efficacement les données produites par une simulation ? Dans certains domaines comme l'économie, le traitement des données de simulation est aisé puisqu'il s'appuie sur les mêmes outils statistiques que ceux utilisés sur le terrain pour mesurer et analyser les phénomènes réels. Dans le cas des simulations multi-agents, souvent utilisées pour simuler des phénomènes biologiques ou sociologiques, la simulation est un des premiers outils utilisés pour analyser le phénomène réel, car les outils classiques achoppent à accomplir cette tâche. Mais, les particularités, notamment dimensionnelle, des données produites les rend difficilement valorisables par les statistiques classiques. Les outils de fouille de données classiques, et leurs extensions à des données multidimensionnelles deviennent indispensables dans ce genre de cas.

Nous allons présenter dans la suite les spécificités des données de simulation et les problématiques qui s'y rattachent. Nous dresserons d'abord un rapide historique de la fouille de données et de l'extraction des connaissances à partir des données. Nous présenterons ensuite les particularités des données de simulation et les difficultés d'utilisation des outils classiques de fouille sur ce type de données. Une partie concernant les méthodes et outils de stockage des données multidimensionnelles précédera celle traitant de la fouille à proprement parler de données spatiales et temporelles.

3.1 Historique, objectif et principe général de la fouille de données

Chacun de nous s'est depuis longtemps parfaitement adapté aux formidables volumes de données qui nous sont accessibles via l'ordinateur et aux outils pour les parcourir. Pourtant, il n'y a encore qu'une trentaine d'années les volumes de données n'avaient rien de comparable et c'est avec joie que l'on savourait les quelques octets que les progrès technologiques ajoutaient aux supports de stockage. Aujourd'hui, il n'est pas rare à titre individuel de disposer de volumes de stockage avoisinant le Tera³⁹. On estime que le volume de données numériques stocké dans le monde double tous les 20 mois (Lyman and Varian, 2003). La question de l'augmentation du volume de données en a rapidement posé deux autres : comment gérer efficacement et se retrouver dans ces volumes de données ? comment valoriser ces données pour n'y découvrir que l'information pertinente ? À la première question, les bases de données et datawarehouses ont rapidement répondu, permettant une organisation optimale des données. Des outils de gestion adaptés ont également fait leur apparition. Pour la deuxième, des solutions toujours plus efficaces se sont développées dans différents domaines de la recherche informatique depuis 1990, regroupées sous la dénomination de méthodes de fouille de données (*data mining* en Anglais)

³⁹À titre informatif, on estime que 10To-octets ($1 \text{ To} = 10^{12} = 1\,000\,000\,000\,000$ octets) permettent de contenir l'ensemble des ouvrages d'une bibliothèque de l'envergure de la Bibliothèque Nationale Française.

ou Extraction des Connaissances à partir des Données (ECD).

D'une manière générale, l'ECD consiste à partir d'un grand volume de données à présenter à l'utilisateur des informations plus synthétiques ou des données particulièrement pertinentes dans le contexte qui l'intéresse. On parle d'extraction de *connaissances* car à partir d'un volume initial de données inintelligibles et inexploitable en l'état, le processus de fouille fournit des informations explicatives permettant à l'utilisateur d'appréhender plus précisément les phénomènes à l'origine de la production de ces données : on parle donc de *connaissances* ou de *pépites* (en analogie avec le terme *data mining*). Cette recherche peut être conduite de manière totalement non supervisée sans postulat de départ sur le caractère exogène ou endogène⁴⁰ des variables ou supervisée où sont prédéfinies a priori les variables explicatives et celles à expliquer. Plusieurs méthodes existent aujourd'hui pour mener à bien cette fouille, de méthodes purement calculatoires (Agrawal and Srikant, 1994) à des méthodes beaucoup plus visuelles (Blanchard, 2005). Toutes ces méthodes partagent en général les mêmes étapes (Fayyad et al., 1996; Blanchard, 2005) :

- **le pré-traitement** : La plupart du temps, on travaille sur des données brutes présentant donc certaines imperfections. Il s'agit donc de les rendre propres à la « consommation » par le processus d'ECD : sélection des données, élimination des doublons, élimination des valeurs aberrantes, gestion des valeurs manquantes, transformation des variables, création de nouvelles variables, ...
- **la fouille de données**⁴¹ : Cette étape, cœur du processus d'ECD, consiste à identifier les motifs structurant les données, découvrir les modèles explicatifs ou prédictifs des données,
- **le post-traitement** : Les connaissances extraites nécessitent généralement une petite passe cosmétique pour permettre à l'utilisateur de les exploiter, de les interpréter et de les valider. La plupart du temps le volume de connaissances (ou de règles) trouvé est important et nécessite encore un effort d'organisation et de sélection soit automatique, soit conduit par l'utilisateur.

Les méthodes de fouille de données sont nombreuses. La recherche de règles tentent d'extraire à partir des données des règles d'associations entre attributs, cela s'apparente généralement à rechercher les motifs fréquents dans une base de données. On peut en trouver l'application par exemple dans l'étude du comportement des consommateurs avec la recherche des produits les plus fréquemment achetés ensembles afin d'optimiser leur placement dans les rayons (exemple du panier de la ménagère). Le *clustering* (également appelé classification non supervisée) tente de segmenter automatiquement les données pour les regrouper en ensembles partageant des caractéristiques communes. La classification supervisée poursuit le même objectif une contrainte initiale spécifique sur la formation des classes (Liu and Motoda, 1998). C'est une méthode largement utilisée notamment dans le milieu de la finance pour l'évaluation du risque au crédit.

⁴⁰ Respectivement *explicatif* ou à *expliquer*.

⁴¹ Kodratoff (2001) distingue d'ailleurs le processus d'ECD du processus de fouille, le premier décrivant l'ensemble du cycle de la sélection et la préparation des données à l'exploitation des résultats et surtout utilisé dans le domaine industriel, alors que le second décrit précisément le processus d'extraction et couvre généralement les méthodes pour y parvenir développées dans le domaine universitaire.

3.2 La simulation et la problématique des données multidimensionnelles

A l'instar des phénomènes réels et à la différence de la plupart des cas d'études classiques réductibles à des tableaux *individus* \times *variables* que l'on peut rencontrer en data mining, chaque simulation produit des jeux de données dont la typologie est extrêmement spécifique. Certaines simulations produisent des données transversales classiques (Sauvy, 1979) : données *individus* \times *variables* caractérisant un état à un instant donné. Ces données transversales peuvent être de natures très différentes : variables binaires, données quantitatives, qualitatives, catégorielles, intervalles, etc.. D'autre part, la plupart des simulations produisent des données longitudinales caractérisant l'évolution dans le temps d'un système. On parle alors de *séries chronologiques*⁴² (Delobel, 1983). À la dimension temporelle peut s'ajouter la dimension spatiale dans le cas où le parcours des entités de la simulation est enregistré. Dans le domaine des SIG, de nombreux travaux ont déjà été accomplis pour l'étude des données spatiales, mais l'étude de leur évolution dans le temps est encore un domaine de recherche ouvert. Enfin, ces données spatio-temporelles peuvent correspondre à une itération de la simulation. Afin de disposer d'un échantillon caractéristique de données, on procède la plupart du temps à itération des simulations. Les outils classiques de traitement ne sont plus suffisants pour traiter ce type d'information. Se pose donc la problématique de trouver des outils appropriés pour l'étude de ces données. La figure 3.1 illustre cette problématique.

3.2.1 Les dimensions spatiale et temporelle

La dimension spatiale

La dimension spatiale peut revêtir des caractéristiques variées suivant les domaines étudiant les données. Dans les SIG⁴³, les données sont principalement spatiales et peu dynamiques (on s'intéresse plus à des données géographiques ponctuelles qu'à leur évolution dans le temps). Les systèmes développés pour traiter les données sont donc principalement axés sur la gestion de données spatiales. Pour l'étude de phénomènes mobiles (dynamique des fluides, étude de processus migratoires, suivi d'objets mobiles en temps réel), ces systèmes doivent également intégrer l'aspect dynamique des données et donc croiser les dimensions spatiales et temporelles. D'autre part, les objets étudiés peuvent eux-mêmes présenter des caractéristiques spatiales différentes, il peut s'agir de formes géométriques, d'objets ponctuels⁴⁴ ou d'objets subissant des déformations au cours du temps (feu de forêt, fluide, solide déformable). Dans tous les cas précités, il s'agit d'étudier des données décrivant les trajectoires d'objet via leur coordonnées dans l'espace. On peut également rencontrer la dimension spatiale sous une forme sémantique où il s'agit de décrire les relations des objets les uns par rapport aux autres. Une position ne sera

⁴²*Time series*, en Anglais.

⁴³Système d'Information Géographique, un domaine où la recherche a été intense et a produit de nombreux résultats.

⁴⁴Un objet est dit ponctuel en physique quand il est assimilable à un point. Il s'agit en fait d'une simplification facilitant l'étude de l'objet et de sa trajectoire.

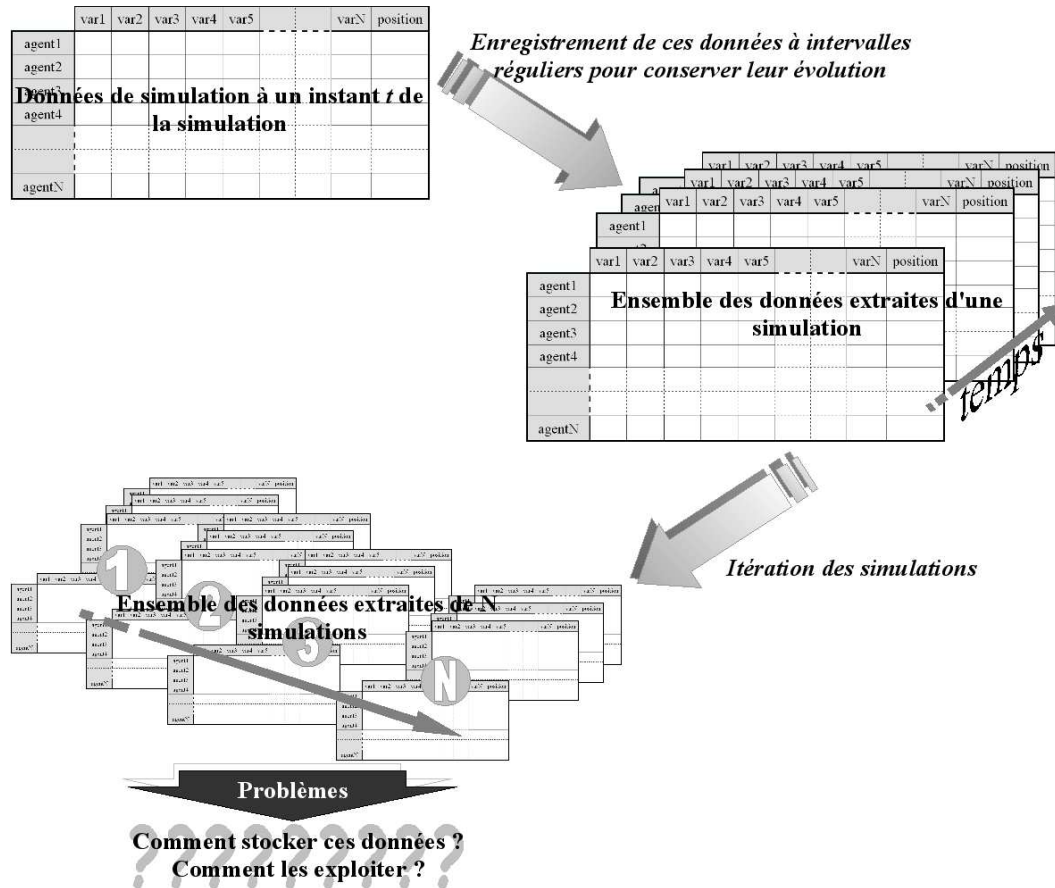


FIG. 3.1 – Problématique de l'analyse des données de simulation

alors plus décrite par ses coordonnées mais par sa relation de voisinage par rapport à un autre objet. D'une manière générale, si les recherches sur les données spatiales statiques ont fourni de nombreux résultats, c'est beaucoup moins le cas pour les données spatiales dynamiques.

La dimension temporelle

Il est intéressant de constater que si les recherches consacrées à la dimension spatiale sont pour la plupart issues de la physique et de la géographie, la dimension temporelle des données a plutôt suscité l'intérêt des économistes et des statisticiens. L'étude de l'évolution d'indicateurs économiques au cours du temps est toujours un sujet d'étude important. C'est donc dès les années 70 avec les premiers SGBD que les recherches sur le sujet sont apparues. La plupart de ces travaux se basent sur des logiques temporelles que nous présenterons dans la partie 3.3.2.

La difficulté de ces recherches réside dans les structures multiples avec lesquelles nous pouvons appréhender le temps. Le temps peut en effet être considéré comme linéaire, ramifié (ou hypothétique) ou cyclique comme le montre la figure 3.2. Dans le cas d'une structure ramifiée, on pourra avoir une ramification dans le futur (un seul passé, plusieurs futurs : ordre linéaire à gauche), dans le passé (plusieurs passés, un seul futur : ordre linéaire à droite)⁴⁵ ou dans les 2 sens (aucun ordre). Le temps peut être fini ou non (typiquement, dans une structure cyclique on a ni début, ni fin, donc un temps infini). Le temps peut être discret (entre deux instants⁴⁶ t , il y a un nombre fini d'instants, temps isomorphe aux entiers), dense (entre deux instants t , il y a un nombre infini d'instants, isomorphe aux rationnels) ou continu (entre deux instants t , il y a un nombre infini d'instants, temps isomorphe aux réels). Enfin, on peut avoir une datation par points (le temps est représenté par une date) ou par intervalles (le temps est représenté par un intervalle de datation dont la durée peut être nulle) (Bernadet, 2002).

Dans le cas d'une simulation, les données seront recueillies épisodiquement au cours du déroulement de la simulation. La structure temporelle correspondante est donc généralement du type discrète, bornée (début et fin de la simulation). Une unique itération d'une simulation produit à partir d'un passé unique un futur unique, nous nous trouvons donc dans le cas d'une structure temporelle linéaire à l'échelle de l'itération de la simulation. À l'échelle supérieure, si on considère de multiples itérations de la simulation ayant toutes un passé commun (avec des paramètres d'initialisation identiques), on aura donc une structure ramifiée à droite à partir du début de la simulation correspondant aux multiples issues que pourra produire la simulation à partir d'une situation de départ identique. Cette multiplicité des futurs à partir d'un même passé est liée au caractère non-déterministe d'une simulation : la connaissance de l'instant ne permet pas de prédire ce que sera le futur, chaque instant peut donc a priori donner naissance à une multitude de branches temporelles. Enfin, il est possible que plusieurs simulations avec des paramètres de départ différents aboutissent à un moment donné à des états du système

⁴⁵On parle d'ordre linéaire à gauche (resp. à droite) si pour toute date t_1 , l'ensemble de ses prédécesseurs (resp. successeurs) est totalement ordonné ($\forall (t_1, t_2, t_3) \in T^3 [(t_2 < t_1) \& (t_3 < t_1)]$ (resp. $[(t_2 > t_1) \& (t_3 > t_1)]$) $\Rightarrow [(t_2 < t_3) \vee (t_2 = t_3) \vee (t_2 > t_3)]$).

⁴⁶L'instant peut également être nommé *chronon*. Il correspond à une durée minimale de temps de longueur fixée (Ariav, 1986; Clifford and Rao, 1987)

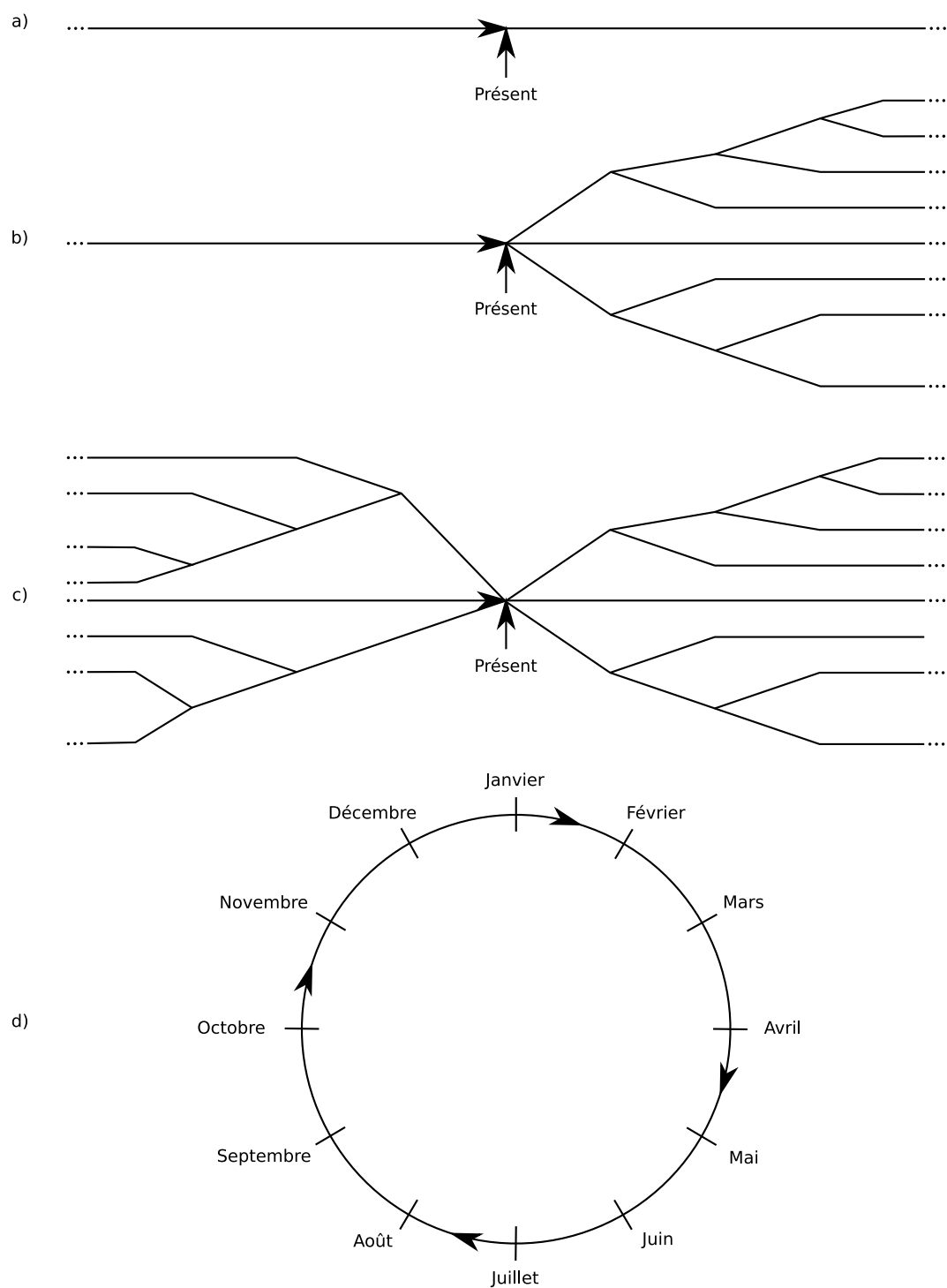


FIG. 3.2 – Structures temporelles a) linéaire, b) et c) ramifiée et d) cyclique.

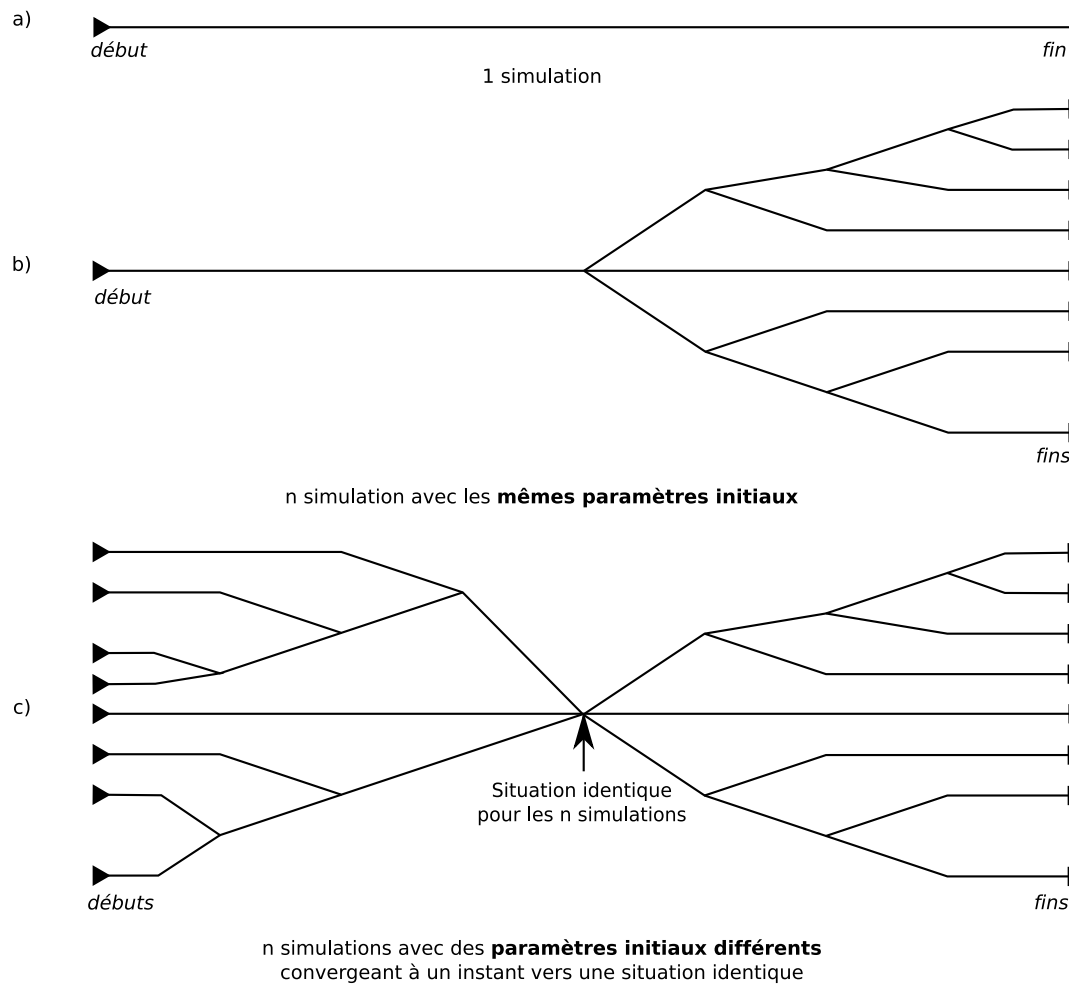


FIG. 3.3 – Structures temporelles pour 1 à n simulations.

identiques⁴⁷ On pourra donc se trouver face à une structure de temps ramifiée des deux côtés relativement à l'instant où les différentes simulations ont convergé (point de similarité). La figure 3.3 synthétise l'ensemble des cas possibles.

⁴⁷La notion d'*identique* est ici subordonnée à la définition d'une mesure de similarité entre deux états de la simulation. Elle permet dans le cadre d'itérations multiples de la même simulation ou dans celui de simulations différentes partageant des caractéristiques communes d'établir un indice de similarité entre les résultats que celles-ci peuvent produire à un instant donné. Cette mesure peut être plus ou moins proche de la mesure d'égalité mathématiques en fonction des caractéristiques de chaque type de simulation (besoin d'une mesure de similarité forte (au sens égalité) dans des simulations physiques, et d'une mesure de similarité faible (au sens ressemblance) dans des simulations sociales).

	Par points	Par intervalles
Cardinalité	Proportionnelle à la durée de l'observation	Proportionnelle aux nombres de changements
Contraintes d'intégrité	Dans le schéma	À la charge des applications
Requêtes	Assez naturelles	Complexes à exprimer

TAB. 3.1 – Propriétés des SGBDs temporels avec datation par points et par intervalles.

3.3 Techniques et outils de stockage de données multidimensionnelles

3.3.1 La notion de temps et les logiques temporelles

Historiquement, les logiques temporelles font partie des premiers travaux à avoir donné un cadre de représentation formel aux données temporelles dans l'informatique. Elles sont à l'origine de la plupart des concepts que l'on peut rencontrer dans les SGBD temporels actuels. Les SGBD temporels ne sont pas la seule solution aujourd'hui pour stocker des données temporelles, mais présentent un ensemble de caractéristiques très intéressantes dans l'hypothèse d'une fouille de données.

3.3.2 Les logiques temporelles

Logiques temporelles réifiées ou avec datation

Dans les logiques temporelles avec datation, on distingue les logiques temporelles avec datation par points et par intervalles. La datation par points correspond à la représentation du temps par une succession de dates ponctuelles, alors que la datation par intervalles représente le temps par des intervalles entre deux dates. Il existe plusieurs logiques par intervalles dont celle de Allen (1983), de Mac Dermott (1982) et de Shoham (1987). La logique par points implique des relations d'ordre sur les éléments alors que la logique par intervalles implique des relations d'ordre sur les intervalles disjoints et les relations de type ensembliste d'inclusion et de chevauchement. Dans un SGBD temporel, la logique avec datation par points associe à un tuple une ou plusieurs dates auxquelles il est vrai alors que la logique avec datation par intervalles lui associe le ou les intervalles de temps pendant lesquels il est vrai (cf. figure 3.4). Le tableau 3.1 montre les avantages et les inconvénients de l'utilisation de chacune de ces deux logiques dans l'élaboration d'un SGBD relationnel temporel (Fauvet, 2003).

Logiques temporelles modales

Les logiques modales considèrent uniquement les relations entre les dates ou les intervalles. On distingue les logiques modales de points et d'intervalles, linéaires ou arborescentes, discrètes ou continues (cf. tableau 3.2). Elles correspondent à un niveau d'abstraction du temps plus élevé

Par points	Par intervalles
<p>Le niveau de stress de l'avatar 1 est : 45 à 0'00", 0'30", 1'00", 60 à 1'30", 2'00", 45 à 2'30".</p> <p>stress/avatar1, 45, 0'00") stress/avatar1, 45, 0'30") stress/avatar1, 45, 1'00") stress/avatar1, 60, 1'30") stress/avatar1, 60, 2'00") stress/avatar1, 45, 2'30")</p>	<p>Le niveau de stress de l'avatar 1 est : 45 de 0'00" à 1'30", 60 de 1'30" à 2'30", 45 à 2'30".</p> <p>stress/avatar1, 45, [0'00", 1'30"] stress/avatar1, 45, [1'30", 2'30"] stress/avatar1, 45, [2'30", 2'30"]</p>

FIG. 3.4 – Datation par points et par intervalles.

Logiques temporelles propositionnelles							
Réifiées		Modales					
Points	Intervalles	Intervalles				Intervalles	
		Linéaires		Arborescentes		Linéaires	
		Discrètes	Continues	Discrètes	Continues	Discrètes	Continues

Peu utilisées car trop complexes.

TAB. 3.2 – Taxonomie des logiques temporelles (von Bertalanffy, 1973).

que pour les logiques avec datation, et fournissent des prédicats du type « les employés feront une pause », « il a toujours été rigoureux », « la machine fonctionnera jusqu'à ce qu'il n'y ait plus de consommable », ...Elles permettent d'exprimer des requêtes complexes sur les bases de données temporelles.

3.3.3 L'intégration du temps et de l'espace dans les SGBD relationnels et objets

La difficulté de gérer les bases de données spatio-temporelles résident bien plus dans le temps que dans l'espace. En effet, la dimension spatiale peut la plupart du temps se décliner en un ou plusieurs attributs supplémentaires de la table associée au caractéristiques d'une entité d'un système afin de mémoriser sa position. On se contente souvent de stocker ses coordonnées dans le système. Comme nous l'avons dit auparavant, ceci est bien entendu valable pour les entités assimilables à un point. Dans tous les cas, la technique reste la même mais le type de donnée stockée pour caractériser la position peut être plus ou moins complexe qu'un couple de coordonnée (variable qualitative décrivant par exemple l'inclusion de l'entité dans une zone géographique, équations physiques caractérisant la forme de l'entité). Dans les faits, on a donc un tableau *individus* × *variables(dont temps)*. Les outils de fouille de données classique sont donc adaptés au traitement de ce type de données. A contrario, la dimension temporelle contraint à une réplification de l'ensemble des variables pour chaque instant ou intervalle que l'on désire

stocker. On a donc un tableau *individu* \times *variable* \times *temps*. Plus qu'une nouvelle variable, il s'agit donc d'une nouvelle dimension des données que peu d'outils de fouille savent appréhender.

Le temps dans les SGBD relationnels

Depuis la fin des années 80, les travaux de recherche (Adiba et al., 1987; Jensen et al., 1993, 1994; Snodgrass, 1994; Canavaggio, 1997; Quidoz, 1997; Dumas Menjivar, 2000), les conférences (TAI, 1987; TIM, 1995) et les ouvrages (Tansel et al., 1993; Etzion et al., 1998) consacrés au sujet se sont multipliés. Les concepts sont aujourd'hui relativement stabilisés, spécialement dans le cadre des SGBD relationnels. La norme TSQL2 a défini des standards au niveau des types temporels et des langages de requêtes (Snodgrass, 1995; TSQ, 2004).

Il s'agit d'abord de distinguer les attributs dont on désire garder l'évolution dans le SGBD, des autres considérés comme statiques. Ces derniers sont mis à jour à chaque modification de leur valeur, alors que pour les premiers on va conserver un historique de l'ensemble de leur valeur dans le temps. Pour les modèles basés sur une datation par points, à chaque mise à jour de la base de données, il s'agira de joindre à l'attribut, s'il a été modifié, détruit ou créé, un marqueur temporel correspondant à la date de cette opération. Pour les modèles basés sur une datation par intervalles, on procède de la même façon avec des intervalles de temps, les bornes de ces intervalles correspondant aux dates des opérations sur l'élément. Le temps peut-être pris en compte soit au niveau des n-uplets (c'est le cas de TSQL2) : si un attribut du n-uplet est modifié on met un marqueur temporel pour l'ensemble du n-uplet, soit au niveau des attributs (c'est le cas de HRDM (Clifford and Crocker, 1993)) où le marqueur n'est déposé que sur l'attribut concerné (Canavaggio, 1997). La figure 3.3.3 illustre ces différents cas.

La complexité se situe surtout au niveau des langages de requêtes qui doivent pouvoir fournir des moyens d'interrogation de la base de données assez riches. De la requête la plus simple, « Quelle était la valeur de stress à 0'30" ? » à des plus complexes du type « Quels attributs ont subi des modifications quand l'attribut stress était supérieur à 65 ? », le langage de requête doit fournir des mécanismes transparents pour l'utilisateur de conversion des différents formats d'expression du temps, de traitement des différents niveaux de granularité du temps, de calcul de durée, etc. . Il existe de nombreuses extensions temporelles de l'algèbre relationnelle permettant de faire des projections temporelles, des sélections, des restrictions, des produits cartésiens et des opérations ensemblistes d'union, d'intersection et de différences temporels. Le temps de réponse doit également rester raisonnable malgré l'augmentation de la complexité.

Le temps dans les SGBD objets

Dans les SGBD objets, les travaux sont plus hétérogènes. L'équipe STORM⁴⁸ du laboratoire LSR de l'IMAG a mené de nombreux travaux dans ce sens. Le plus important est sans doute le projet TEMPOS (Canavaggio, 1997; Dumas Menjivar, 2000) dont le but était d'étendre le

⁴⁸Pour *Structure et Temporalité des Objets Réactifs Multimédia*.

(a) Datation par points au niveau des tuples				(b) Datation par intervalles au niveau des tuples			
Entité	Stress	Fatigue	Date	Entité	Stress	Fatigue	Date
Individu1	40	50	0'00"	Individu1	40	50	[0'00", 0'30"]
Individu1	40	55	0'30"	Individu1	40	55	[0'30", 0'45"]
Individu1	60	55	0'45"	Individu1	60	55	[0'45", 1'00"]
Individu1	70	60	1'00"	Individu1	70	60	[1'00", 1'30"]
Individu1	70	65	1'30"	Individu1	70	65	[1'30", 2'00"]
Individu1	60	65	2'00"	Individu1	60	65	[2'00", -]
Individu2	60	55	0'00"	Individu2	60	55	[0'00", 0'45"]
Individu2	40	50	0'45"	Individu2	40	50	[0'45", 1'30"]
Individu2	50	50	1'30"	Individu2	50	50	[1'30", 2'00"]
Individu2	55	50	2'00"	Individu2	55	50	[2'00", -]

(c) Datation par points au niveau des attributs			(d) Datation par intervalles au niveau des attributs		
Entité	Stress	Fatigue	Entité	Stress	Fatigue
Individu1	40 [0'00"]	50 [0'00"]	Individu1	40 [0'00", 0'45"]	50 [0'00", 0'30"]
	60 [0'45"]	55 [0'30"]		60 [0'45", 1'00"]	55 [0'30", 1'00"]
	70 [1'00"]	60 [1'00"]		70 [1'00", 2'00"]	60 [1'00", 1'30"]
	60 [2'00"]	65 [1'30"]		60 [2'00", -]	65 [1'30", -]
Individu2	60 [0'00"]	55 [0'00"]	Individu2	60 [0'00", 0'45"]	55 [0'00", 0'45"]
	40 [0'45"]	50 [0'45"]		40 [0'45", 1'30"]	50 [0'45", -]
	50 [1'30"]			50 [1'30", 2'00"]	
	55 [2'00"]			55 [2'00", -]	

TAB. 3.3 – Intégration des historiques dans le modèle relationnel.

SGDB objet O_2 avec des fonctionnalités temporelles. Les caractéristiques sont à peu près les mêmes que celles en relationnel. Le temps peut cette fois-ci être pris en compte au niveau des objets ou au niveau des attributs (de la même façon qu'en relationnel au niveau des tuples ou des attributs) (Canavaggio, 1997).

Dans les modèles objets, les requêtes bénéficient de la puissance d'expression des langages objets et les adaptations nécessaires du modèle sont moins lourdes à mettre en place. Néanmoins, on rencontre toujours le même problème de complexité tant au niveau du calcul qu'au niveau des requêtes elles-mêmes qui peuvent être assez difficiles à exprimer malgré la puissance d'expression du langage.

3.4 Data mining temporel

Nous avons vu les solutions permettant d'intégrer le temps et l'espace dans un SGBD. Le problème est maintenant le même que celui qu'on rencontre avec la plupart des bases de données : que faire de ces données et quelles informations apportent-elles ? Pour répondre à cette question, le domaine de l'extraction de connaissances à partir des données connaît un essor important depuis une quinzaine d'années. Des techniques très efficaces ont été développées pour s'appliquer à des bases de données classiques *individus* \times *variables* dès le début des années 80. Pour les données spatiales et temporelles, les recherches ont d'abord concerné des domaines très spécifiques tels que les prévisions météorologiques, l'analyse des tendances boursières ou encore les recherches portant sur les SID (Box et al., 1994; Chatfield, 1996). Les efforts menés au cours du milieu des années 90 pour la reconnaissance de la parole ont ensuite fourni des résultats intéressants en matière de classification et d'étude des correspondances entre séries temporelles (Juang and Rabiner, 1993; O'Shaughnessy, 2003).

La fouille de données spatio-temporelles est quant à elle beaucoup plus récente et n'est ni soumise aux mêmes contraintes, ni ne poursuit les mêmes objectifs (Laxman and Sastry, 2006). La première différence concerne la nature des données et la façon dont elles sont connectées. La plupart du temps, les méthodes de fouille de données doivent traiter des volumes de données bien trop grand pour les méthodes classiques de fouille de données séquentielles. D'autre part, les données peuvent être qualitatives ou quantitatives rendant les techniques statistiques caduques. De plus, le processus de collecte des données n'est, généralement, pas maîtrisé⁴⁹ et les données sont souvent collectées pour un tout autre but que la fouille. Enfin, les objectifs de l'ECD sont plus divers que ceux de la fouille de séries temporelles (la prédiction et le contrôle). Il ne s'agit pas d'obtenir un modèle exact d'évolution des données, mais plutôt d'y déceler des informations rares et particulièrement intéressantes. Nous reviendrons sur cette distinction un peu plus loin dans ce chapitre.

Roddick et al. (2001) ont dressé une vue d'ensemble riche et incontournable des travaux parus sur le data mining temporel sur laquelle nous nous sommes basés. Laxman and Sastry (2006) ont plus récemment proposé une liste complémentaire de travaux plus récents. Les pistes explo-

⁴⁹Ce dernier point concerne peu les données de simulation puisque pour celles-ci, on peut justement adapter la génération des données aux besoins de la fouille.

rées et que nous allons présenter dans la suite sont la plupart du temps des extensions de ce qui se fait déjà pour les bases de données classiques ou des techniques développées spécifiquement pour les données temporelles, spatiales et multidimensionnelles. Des techniques interactives de fouille de données existent également pour ce type de données. Nous avons choisi de ne présenter que les techniques de data mining qui doivent ensuite s'intégrer dans un processus plus large d'ECD.

3.4.1 Préliminaires

Nous présentons dans cette partie des prérequis nécessaires à la suite de ce chapitre : la notion de série chronologique et les notions afférentes, la différence entre la notion de motif et de modèle dans la fouille de données et enfin la problématique d'ECD sur des données issues de la simulation.

Notions relatives aux séries chronologiques

Les séries chronologiques sont particulièrement étudiées en statistiques. C'est pourquoi il existe déjà de nombreux outils que n'ont pas manqué d'employer les méthodes de data mining temporel. Ces outils reposent en partie sur l'étude de propriétés caractéristiques des séries chronologiques. Ainsi, une série est décomposable en 3 sous-séries : sa tendance, son cycle (ou périodicité) et ses variations aléatoires (ou bruit) (Benavent, 1999). La figure 3.5 illustre cette décomposition. La tendance est en général la composante la plus importante d'une série et celle qui va nous intéresser le plus pour étudier l'évolution des caractéristiques des entités du système. On distingue naturellement trois types primaires d'évolution : croissance, stagnation ou décroissance. Le cycle d'une série chronologique est intéressant pour les phénomènes périodiques. On l'a illustré sur la figure mais il est peu probable qu'on rencontre des séries comportant des cycles dans la simulation. Il faut néanmoins être vigilant et être capable de les détecter s'ils sont présents, car un phénomène cyclique est très intéressant car prédictible. Les variations aléatoires (ou stochastiques) correspondent à des micro-évolution de la série. Elles présentent peu d'intérêt car ne sont pas très riches en contenu informationnel. Il est probable que le processus de simulation engendre de telles variations qui correspondent à ce qu'on rencontre dans le monde réel. En data mining, il est intéressant de ne travailler que sur la tendance parce qu'elle traduit un phénomène général, même si les procédures de lissage permettant de l'isoler sont parfois difficiles à mettre en œuvre et peuvent altérer le contenu informationnel de la série si les variations aléatoires sont dues à des causes qu'il serait intéressant de connaître.

La différence entre modèle et motif

Les types de structure que les algorithmes de fouille recherchent peuvent être classés de multiple façons (Han and Kamber, 2001; Witten and Frank, 2000; Hand et al., 2001). Par exemple, il est souvent judicieux de catégoriser les sorties des algorithmes en modèles et motifs (Hand et al., 2001).

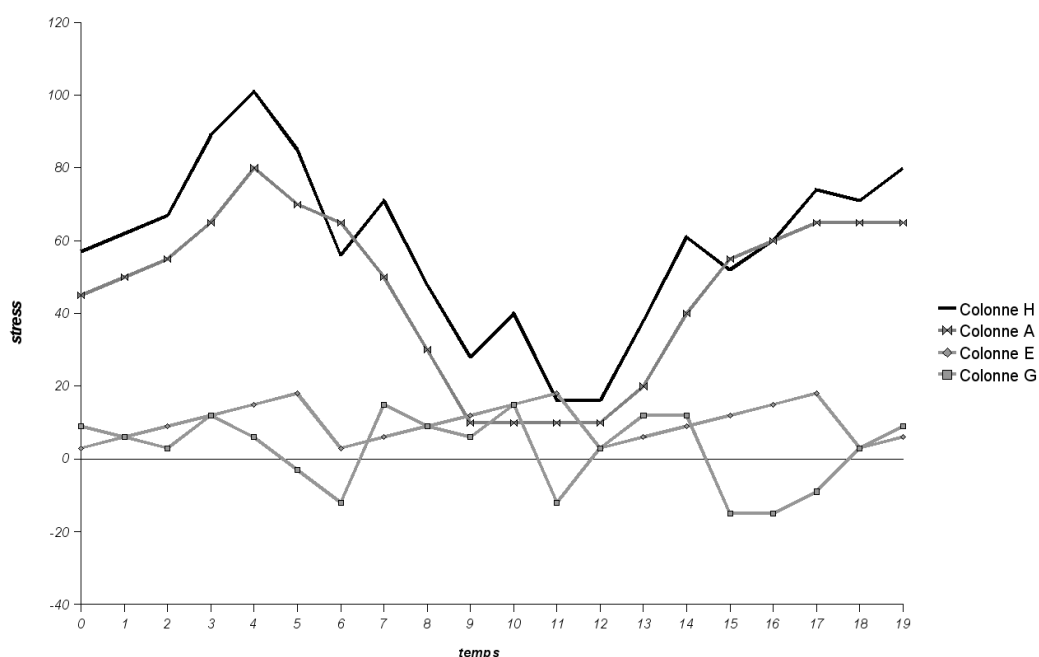


FIG. 3.5 – Décomposition d'une série chronologique..

Un *modèle* est une représentation générale de haut niveau et souvent abstraite des données. On distingue généralement les modèles prédictifs des modèles descriptifs. Les modèles prédictifs sont utilisés à des fins de prévision ou de classification alors que les modèles descriptifs sont utilisés pour obtenir une vue synthétique des données. Les méthodes d'auto-régression (Box et al., 1994), par exemple, sont utiles pour prédire l'évolution d'une série temporelle en tenant compte de son passé. Les chaînes de Markov (Juang and Rabiner, 1993; O'Shaughnessy, 2003) sont également largement utilisées à des fins de prédiction. A contrario, les spectrogrammes (obtenus à partir d'une analyse temps-fréquence des séries temporelles) et les méthodes de clustering sont des techniques descriptives utilisées à la fois pour la synthèse et pour la visualisation des données.

À la différence des modèles, le *motif* est une structure locale qui constitue d'un échantillon réduit des données. Dans les séries temporelles, les creux et les pics sont de bons exemples de motifs pouvant présenter un intérêt. Ainsi, dans les tâches de surveillance, la fouille de séries temporelles grâce à des expressions régulières pour identifier la survenue d'événements précis ou, en bio-informatique, celle des gènes dans l'ADN constituent deux exemples de l'application de la recherche de motifs.

Cette distinction présente un intérêt dans la catégorisation et dans le choix d'une méthode de fouille de données. Néanmoins, certaines méthodes ne peuvent aisément être classées dans l'une ou l'autre de ces catégories. Il peut en effet être parfois intéressant de coupler à une recherche de motifs, une analyse statistique des résultats obtenus, ceci afin d'obtenir une meilleure interprétation des informations apportées par la fouille.

Problématique

Il va être nécessaire de trouver des moyens permettant d'analyser de manière pertinente les données produites par les simulations. D'une part, il va falloir être capable de fournir une vue synthétique de l'évolution d'une ou plusieurs simulations afin qu'elles soient facilement analysables. C'est du domaine de la visualisation mais également de la fouille de données. D'autre part, il va être nécessaire de dégager des règles d'évolution caractéristiques par rapport à l'ensemble des simulations. Par exemple, si dans de multiples simulations, on constate que la rencontre de l'individu α avec l'individu β sur le poste 1 quand α est dans un état de stress et β est dans un état de fatigue conduit à une situation conflictuelle, cette règle doit être mise en évidence. On a donc d'une part la prise en compte des caractéristiques de chaque individu, de leur situation spatiale et de l'évolution temporelle de la situation.

Plusieurs techniques vont donc devoir être mises en œuvre :

- les techniques de fouille de données classiques sur chaque capture à un instant donné de l'état d'une simulation donnée, c'est à dire sur chaque table individus x variables, nous ne parlerons pas de ces techniques ici,
- l'extraction de règles d'association au niveau temporel permettant de produire des règles de type « si cause à instant t, alors effet à instant $t+\delta$ »,
- la classification de séries chronologiques,
- la recherche de motifs entre plusieurs séries et plusieurs simulations permettant de caractériser des évolutions qu'on retrouve fréquemment,
- l'extraction d'information des données spatiales.

3.4.2 Les tâches classiques d'étude de données temporelles

On a vu que les bases de données temporelles étaient elle-même encore un domaine où la recherche et l'adoption de standards avait du chemin à faire. C'est dans ce contexte, et donc avec l'utilisation de ce qui se fait actuellement au niveau des bases de données temporelles, que se situe les techniques de fouille de données temporelles que nous allons présenter. Certaines publications relatives à l'ECDT⁵⁰ insistent d'ailleurs sur la nécessité de définir un modèle pour les données temporelles (Lin et al., 2002b). Nous allons plus particulièrement développer les techniques relatives aux séries chronologiques⁵¹, structure qui caractérise nos données.

Les méthodes d'ECD sont utilisées dans nombre de domaines très disparates. Pourtant, les objectifs poursuivis par ces méthodes peuvent être classés en quelques grands groupes généraux (Han and Kamber, 2001; Hand et al., 2001). Dans le cas plus particulier de la fouille de données temporelles, ces tâches peuvent se distinguer de la sorte : la prédiction, la classification, le clustering, la recherche et l'extraction et la découverte de motifs. Cette catégorisation ne se veut ni unique, ni exhaustive. Elle a pour seul objectif d'aider à classer les nombreuses

⁵⁰Extraction de Connaissances à Partir de Données Temporelles, ou *Knowledge Discovery in Temporal Databases*, en Anglais.

⁵¹Les séries chronologiques de la bourse, où la prédiction du cours des actions est un enjeu financier crucial, sont particulièrement étudiées en data mining temporel.

techniques existantes du domaine. Les quatre premières catégories ont déjà fait l'objet de nombreux travaux dans les domaines traditionnels des séries temporelles et de la reconnaissance de motifs, alors que les algorithmes de recherche de motifs dans les bases de données sont plus récents et plus spécifiques au seul domaine de la fouille de données. Nous allons donc dans un premier temps lister le principe et les travaux remarquables dans les domaines de la prédiction, la classification, le clustering, la recherche et l'extraction, puis dans un second temps, détailler plus précisément les différentes recherches menées dans le domaine de la découverte de motifs.

Prédiction

L'objectif de la prédiction sur les séries temporelles est de construire un modèle permettant de prévoir l'évolution des données dans le futur avec un maximum de fiabilité. Les prévisions météorologiques sont sans doute le premier exemple qui vient à l'esprit. Les premiers travaux sur le sujet sont attribués à Yule (1927). Il convient ici de distinguer les séries considérées comme stationnaires pour lesquelles les méthodes d'auto-régression s'avèrent particulièrement adaptée (Box et al., 1994; Chatfield, 1996; Hastie et al., 2001) des séries non-stationnaires pour lesquelles des méthodes plus spécifiques ont été développées. Parmi celles-ci, on peut citer les méthodes ARMA (AutoRegressive Moving Average) et ARIMA (AutoRegressive Integrated Moving Average) utilisées dans de nombreuses applications économiques et industrielles pour obtenir des modèles linéaires non-stationnaires quand les différences entre des termes successifs peuvent, elles, être considérées comme stationnaires. Une autre méthode consiste à fragmenter la série en sous-séries stationnaires et à rechercher des modèles sur ces parties. Dans le cas où le modèle recherché n'est pas linéaire, les réseaux neuronaux ont prouvé leur efficacité.

Classification

La classification est un processus supervisé ayant pour objectif d'assigner un élément à une classe parmi un ensemble de classes prédéfinies. La classification regroupe de nombreuses techniques classiques d'ECD mais n'a paradoxalement pas fait l'objet de beaucoup de recherches en data mining temporel. Ces méthodes sont largement utilisées dans les tâches de reconnaissance : reconnaissance de la parole (Juang and Rabiner, 1993; O'Shaughnessy, 2003; Gold and Morgan, 2000), reconnaissance de l'écriture manuscrite (Kundu et al., 1988; Tappert et al., 1990; Nag et al., 1986), reconnaissance des mouvements (Darrell and Pentland, 1993; Yamato et al., 1992; Starner and Pentland, 1995)), dissociation au sein des bruns d'ADN les zone codant les gènes des zone non expressives, etc.

Un opérateur de classification a été proposé par (Keogh and Pazzani, 1998). Cet opérateur permet de retourner une série dont la tendance est un compromis entre deux séries en entrée. Le principe est d'itérer le processus sur des séries proches afin d'obtenir des modèles généraux correspondant à un ensemble des séries contenues et de définir ainsi une classe. Il est également possible de prendre en compte les séries ne faisant pas partie de la classe (les exemples négatifs) afin d'affiner le modèle général de chaque classe. Le processus correspond encore une fois à celui employé pour les données classiques. On trouve également une autre technique basée

sur les k plus proches voisins (Duda and Hart, 1973). Dans notre cas, cette technique pourrait nous permettre de dégager des grandes tendances comportementales qu'on peut retrouver dans plusieurs simulations, de regrouper des simulations semblables et de fournir via les modèles généraux correspondant à chaque classe une vue synthétique de l'ensemble des simulations.

La phase de classification à proprement parler est souvent précédée d'une phase de recherche de motifs. C'est alors sur les motifs extraits que s'effectue la classification. Au fil du temps, les méthodes de classification se sont également enrichi avec des méthodes de recherche de modèles. Les chaînes de Markov cachées sont utilisées dans la classification des protéines (Baldi et al., 1994; Ewens and Grant, 2001) et la prédiction de l'évolution de marchés financiers (classification des séries temporelles associées) (Tino et al., 2000). Les réseaux neuronaux sont eux aussi également utilisés dans la classification de protéines (Wu et al., 1995) et la classification d'électro-encéphalogrammes (Haselsteiner and Pfurtscheller, 2000).

Clustering

Le *clustering* de séries temporelles consistent à classer de manière non supervisée des séquences temporelles en fonction de leur similarité (et/ou dissimilarité). Dans le clustering, le nombre de classes est spécifié à l'avance ou laissé à la charge de l'algorithme de classification. Le clustering peut-être hiérarchique (arbres de décision) ou non hiérarchique (*k-means*). L'aspect non supervisé du clustering est intéressant à plus d'un titre et est donc utilisé dans de nombreux domaines où on vise à automatiser au maximum le processus de fouille et les résultats fournis.

Ainsi, dans l'analyse de journaux d'activité réseaux, le clustering peut indiquer des habitudes de navigation de différents utilisateurs. Dans l'analyse de données financières, il est intéressant de regrouper les actions qui suivent les mêmes tendances. Le clustering est également utilisé pour regrouper les protéines présentant des similarités dans l'analyse de séquences biologiques (Corpet, 1988; Miller et al., 1999; Naoki et al., 2002). Les méthodes de clustering sont nombreuses : celles à base de modèles (Smyth, 1997; Sebastiani et al., 1999; Law and Kwok, 2000), celles à bases de modèles ARMA (Xiong and Yeung, 2002) ou de chaînes de Markov cachées (Cadez et al., 2000; Alon et al., 2003), celles basées sur l'alignement de motifs (Corpet, 1988; Fadili et al., 2000) ou encore celles à base de mesures de similarités (Schreiber and Schmitz, 1997; Kalpakis and Puttagunta, 2001).

Plus spécifiquement, nous citerons les travaux suivant. Pour effectuer le clustering de séries chronologiques, Smyth (1999, 1997) propose une technique basée sur les chaînes de Markov cachées⁵² et sur l'utilisation de l'algorithme de maximum de vraisemblance⁵³ en associant à chaque classe un modèle général correspondant à une chaîne de Markov cachée. L'algorithme COBWEB (Fisher, 1987) est utilisé par Ketterlin (1997) pour effectuer un clustering hiérarchique de séries chronologiques. Il n'est pas évident de trouver des mesures de distance entre séries chronologiques, nécessaires pour ces méthodes, suffisamment significantes. Les statis-

⁵²HMM pour *Hidden Markov Models*, en Anglais.

⁵³EM pour *Expectation Maximization Algorithm*, en Anglais.

tiques offrent néanmoins des solutions acceptables.

Recherche et extraction

La recherche de séquences temporelles dans les bases de données est une technique particulièrement importante dans les tâches de fouille interactive. Le problème s'apparente à la recherche de séquences (également nommées requêtes) dans des grosses bases de séquences (ou dans une seule longue séquence). Le problème de recherche à partir de requêtes a été largement étudié dans les théories du langage et des automates. Alors que l'extraction de motifs en concordance exacte avec une requête est bien maîtrisée, la situation est légèrement différente quand il s'agit de traiter des concordances partielles ou floues (Wu and Manber, 1992), cas le plus utilisé en data mining. On distingue trois techniques différentes permettant d'extraire des motifs d'une ou plusieurs séries temporelles. Chacune de ces techniques poursuit le même objectif : présenter à l'utilisateur les éléments qui se rapproche le plus de sa requête.

Dans le cas le plus simple, il s'agit de parcourir les données avec une fenêtre de la taille de la requête et de mesurer la similarité entre les sous-séquences ainsi lues et la requête. La pertinence de la méthode est alors fortement liée à la mesure de similarité choisie. Les données étudiées peuvent être de type quantitatives (par exemple dans l'analyse de signaux audio) ou de type qualitatives (par exemple dans l'analyse de séquences de gènes). La distance euclidienne peut s'avérer adaptée pour l'étude de données quantitatives, mais il est des cas où celle-ci se révèle inefficace. Par exemple, dans les signaux audio, des séquences similaires à l'écoute peuvent avoir des distances euclidiennes importantes et inversement (Ghiasi et al., 1995). Dans ces cas, l'utilisation de distances spécifiques effectuant par exemple la transformation des données initiales du domaine temporel au domaine spectral (e.g. distance log-spectral (Gray et al., 1980), distance cepstrale pondérée (Juang and Rabiner, 1993), transformées de Fourier discrète ou en ondelettes discrètes (Wu et al., 2000)) sont souvent préférées. Enfin, quand il s'agit de données qualitatives, une mesure de dissimilarité entre chaque paire de symboles est en général utilisée. On trouve ainsi pour l'étude des séquences génétiques d'acides aminés les méthodes PAM (Gusfield, 1997) et BLOSUM (Ewens and Grant, 2001).

Les méthodes précédentes rencontrent leurs limites lorsqu'il s'agit de mesurer la concordance entre des séquences de longueurs différentes. Dans ce cas, il est nécessaire de procéder à ce qu'on appelle *l'alignement de séquences*. De nombreux travaux proposent des méthodes de déformation temporelle⁵⁴ consistant à appliquer une transformation (dilatation ou contraction) aux séquences temporelles (Kruskal, 1983; Juang and Rabiner, 1993; Gold and Morgan, 2000). Ces transformations consistent la plupart du temps à effectuer un ensemble d'opération d'« édition » (insertion, effacement ou substitution de valeurs ou symboles) (Durbin et al., 1998; Levenshtein, 1966). La somme des coûts de l'ensemble des opérations nécessaires pour aboutir à la concordance d'une sous-séquence et du motif recherché détermine alors la distance entre chacun d'eux.

Enfin, une dernière technique consiste à considérer que deux séquences sont similaires si

⁵⁴DTW pour *Dynamic Time Warping* en anglais.

elles possèdent suffisamment de sous-séquences ordonnées ne se recouvrant pas qui sont similaires. Cette approche est inspirée de l'algorithme *a priori* et a d'ailleurs été développée par Agrawal et al. (1995a). Cette approche est parfois combinée avec une abstraction des données consistant à associer à des caractéristiques de la séquence temporelle (ses variations par exemple) des symboles (Agrawal et al., 1995b). Il devient alors beaucoup moins coûteux de traiter selon cette méthode les séquences de symboles plutôt que celles des valeurs réelles. Keogh and Pazzani (2000) ont ainsi proposé une méthode consistant à associer à une séquence temporelle de longueur n , une approximation par fonction en escalier de longueur $k \ll n$ (Piecewise Aggregate Approximation). La méthode Piecewise Linear Approximation (Keogh et al., 2003) fonctionne de la même manière mais propose une approximation par une fonction affine par morceaux. Ces deux méthodes permettent de diminuer drastiquement les temps de recherche tout en conservant l'information initiale de la séquence.

3.4.3 ECD sur des données temporelles : la découverte de motifs

Les parties précédentes ont déjà introduit la notion de motif, en particulier dans la 3.4.2. Néanmoins, la découverte de motifs présente une différence de taille avec la recherche et l'extraction : aucune requête ne sert de support à cette recherche. Il s'agit ici fouiller les séquences temporelles afin d'y déceler les motifs les plus intéressants. C'est en ce sens que, contrairement aux quatre autres tâches précédemment citées (la prédiction, la classification, le clustering et la recherche et l'extraction), celle-ci, par son caractère exploratoire et non supervisé, est une activité typique de la fouille de données.

La découverte de règles d'associations temporelles est une activité consécutive à la découverte de motifs. Une règle d'association temporelle détermine un lien de causalité temporelle entre un groupe d'événements antérieurs à un ou plusieurs événements. À l'identique des règles d'associations classiques sur des données *individu* \times *variable*, il s'agit donc de produire des règles avec une prémisse et une conclusion liées par une implication conditionnelle. On lie généralement à ces règles des mesures de qualité, précisément les notions de *support* et de *confiance*. Le *support* traduit la représentativité des éléments constituant la règle au sein des données. Ainsi, si on considère une règle $X \Rightarrow Y$ sur un ensemble D , son support est égal à $P(X \cup Y)$, soit $\frac{|X \cup Y|}{|D|}$. La *confiance* traduit la représentativité de la règle elle-même dans les données, c'est à dire la proportion d'éléments vérifiant l'implication. En considérant le même exemple, la confiance de la règle $X \Rightarrow Y$ est égale à $P(Y/X)$, soit $\frac{|X \cap Y|}{|X|}$. On voit donc que le support d'une règle dépend donc directement de la fréquence du motif sur lequel elle repose. Ainsi, la notion de motif intéressant est souvent confondue avec celle de motif fréquent. Si cette assertion est loin d'être fausse, la pertinence d'un motif est fonction d'autres paramètres largement discutés dans des ouvrages traitant des mesures de qualité (Guillet and Hamilton, 2007). En sus, il est à noter que si la notion de support reste pleinement applicable au cas des séquences temporelles, il est par contre plus difficile de définir de manière définitive ce qu'est la mesure de confiance. En effet, les règles d'associations peuvent revêtir des contraintes temporelles qu'il est nécessaire de considérer dans son calcul. Si on reprend l'exemple précédent, $X \Rightarrow Y$ peut

dans certains cas désigner une implication directe⁵⁵ ou plus complexe⁵⁶. Dans ce dernier cas, le calcul de la confiance doit prendre en compte ses contraintes temporelles pour ne dénombrer que les séquences d'évènements s'y conformant.

La découverte de règles d'association temporelles repose sur une extension des techniques déjà éprouvées pour les données non temporelles. La recherche de règles va consister à trouver les motifs fréquents dans les séries chronologiques. On utilise l'algorithme *apriori* (Agrawal and Srikant, 1994) adapté aux données temporelles à cette fin. Cet algorithme, qui consiste dans sa version originale à trouver les *k-itemsets*⁵⁷ fréquents puis à ne chercher de règles que sur ces *itemsets*, est utilisé en ordonnant les itemsets suivant le temps. Par exemple, la suite d'itemsets $\langle \{stress=45, fatigue=30, position=debout\}, \{stress=55, fatigue=60, position=debout\}, \{stress=75, fatigue=80, position=assise\} \rangle$ correspond aux trois suites chronologiques suivantes : $stress = \langle 45, 55, 75 \rangle$, $fatigue = \langle 30, 60, 80 \rangle$ et $position = \langle debout, debout, assis \rangle$ ⁵⁸. L'adaptation la plus importante de l'algorithme de départ est qu'un élément d'un itemset ne pourra être comptabilisé qu'une unique fois dans le calcul de son support. On détermine d'abord les itemsets les plus longs et les plus fréquents, on les indexe. Les séries initiales sont ensuite remplacées par de nouvelles constituées par les itemsets sélectionnés. Ensuite, il s'agit de générer des *k-séries* les plus longues possibles en utilisant l'algorithme *apriori* classique (Agrawal and Srikant, 1995). À partir des séries identifiées, on va pouvoir dégager les règles les plus pertinentes du type « si on est dans l'état A à l'instant t, on sera dans l'état B à l'instant t+1 ».

Une approche différente de recherche de règles dans des séries chronologiques multidimensionnelles telles que nous en aurons est de considérer le temps comme une dimension supplémentaire et d'appliquer des méthodes d'extraction de règles spécifiques aux bases de données multidimensionnelles (Lu et al., 1998). Chaque élément est alors localisé par une adresse qui le situe dans un environnement multidimensionnel. Cela nécessite de redéfinir les concepts de règles d'association, de support et de confiance (Antunes and Oliveira, 2001).

Typologie des recherches de motifs

Les motifs séquentiels La recherche de motifs séquentiels est historiquement la première tâche de data mining effectuée sur des données temporelles. La paternité en revient à Agrawal and Srikant (1995) dans leur publication concernant une extension de leur algorithme *a priori* aux données séquentielles (séquences des transactions effectuées par des consommateurs dans un magasin). À la différence de l'algorithme *a priori* classique, celui-ci s'étend à des séquences de transactions ordonnées dans le temps. Pour le reste, il repose sur le même lemme : toute séquence de taille *n* est non fréquente (*i.e.* avec un support inférieur à un seuil fixé) si elle contient

⁵⁵La séquence d'évènements consécutifs désignée par *X* est directement suivie par *Y*

⁵⁶La séquence *X* peut désigner une suite d'évènements non consécutifs (avec donc des évènements intermédiaires les séparant) et *Y* peut elle-même survenir un certain temps après le dernier évènement de *X*

⁵⁷Un *itemset* est une collection d'items du type « champ=valeur » (ex. $\{stress=45, fatigue=30, position=debout\}$). Un *k-itemset* est un itemset contenant exactement *k* items, l'exemple précédent est un *3-itemset*.

⁵⁸En général, une série chronologique contient des données numériques, mais il est possible d'étendre le concept à d'autres types de données (Quidoz, 1997). Antunes and Oliveira (2001) fait la distinction entre série chronologique et séquence temporelle relativement à cette caractéristique.

une sous-séquence de taille $n - 1$ non fréquente⁵⁹. Le support d'une séquence correspond alors au pourcentage des séquences de transactions de la base contenant cette séquence. Agrawal and Srikant (1995) proposent alors deux approches pour rechercher les séquences fréquentes qu'ils nomment respectivement les algorithmes de comptage total (*count all algorithm*) et comptage partiel (*count some algorithm*). Le premier consiste à appliquer le principe de base de l'algorithme *a priori* en augmentant à chaque itération la taille des motifs recherchés en se basant sur ceux ayant un support suffisant à l'itération précédente. Le second consiste à exploiter le principe de maximalité⁶⁰ en ne travaillant au départ que sur des séquences d'une certaine taille, puis en traitant dans un second temps les séquences restantes.

De nombreux travaux des dernières années sur la recherche de motifs séquentiels ont consisté à améliorer les performances de l'algorithme d'Agrawal and Srikant (1995). Ainsi, Shintani and Kitsuregawa (1998) proposent une parallélisation des algorithmes de fouille. Alors que l'algorithme original de Agrawal and Srikant (1995) nécessite autant de passes que la taille du plus grand motif de la base de données, Zaki (1998) propose une méthode basée sur la théorie des treillis pour fragmenter l'espace de recherche initial et ainsi diminuer drastiquement le nombre de passes nécessaires. Lin and Lee (2003) ont quant à eux proposé une méthode interactive où l'utilisateur fixe itérativement des seuils de support minimaux pour au final obtenir les motifs correspondant au dernier seuil fixé. Cette approche qui consiste à associer l'utilisateur au processus de fouille pour qu'il réduise l'espace de recherche en fixant lui-même des contraintes se retrouve dans plusieurs travaux. Ainsi, Ng et al. (1998) proposent une extension de leur méthode où l'utilisateur définit les contraintes temporelles (intervalles minimum et maximum entre les éléments d'une série temporelle). Garofalakis et al. (2002) ont proposé une méthode baptisée SPIRIT⁶¹ où l'utilisateur contraint le processus de fouille à l'aide d'expressions régulières. Afin d'éviter la problématique de redondance des règles extraites et le surcoût que cela entraîne, Pasquier et al. (1999) ont proposé une méthode, basée sur la fermeture de la connexion de Galois, particulièrement adaptée aux données denses et/ou corrélées. Cette méthode réduit énormément l'espace de recherche en ne s'intéressant qu'aux séquences dites *fermées* pour générer les motifs fréquents, c'est à dire les séquences de taille n pour lesquelles il n'existe aucune séquence de taille supérieure à n ayant le même support. Il convient également de citer les travaux de Yan et al. (2003) et Wang and Han (2004) utilisant eux aussi le principe de fermeture pour améliorer les performances de la fouille.

Les épisodes fréquents Dans le paragraphe précédent, nous avons présenté les travaux concernant la recherche de motifs fréquents dans un ensemble de plusieurs séries temporelles. Une autre approche consiste à rechercher les motifs fréquents au sein d'une unique série temporelle (Mannila et al., 1997). Cette tâche est généralement désignée comme la recherche d'épisodes fréquents. Mannila et al. (1997) utilisent cette méthode pour la découverte d'événements récurrents dans des journaux d'alarmes réseaux. À l'objectif d'analyse déjà présent dans la recherche

⁵⁹Ce principe est appelé *downward closure lemma* en anglais.

⁶⁰Une séquence est dite maximale dans un ensemble de séquences si et seulement si elle n'est contenue dans aucune des autres séquences de cet ensemble.

⁶¹Pour *Sequential Pattern Mining with Regular expression constraints*.

de motifs s'ajoute un second objectif de prévision des événements futurs du phénomène décrit par la séquence objet de l'analyse. La plupart des travaux sur le sujet utilisent des principes très proches de ceux mis en œuvre dans la recherche de motifs. La principale différence réside dans le fait qu'on ne traite pas différents plusieurs séquences différentes et qu'il est donc nécessaire de procéder à une simili-fragmentation de la séquence étudiée. Pour cela, on utilise généralement le principe de fenêtre glissante dont la largeur, la plupart du temps définie a priori, va déterminer la taille maximale des épisodes fréquents extraits. Cette approche a également été utilisée pour étudier des données provenant de chaînes d'assemblage dans les usines (Laxman et al., 2004). Mannila et al. (1997) ont également étendu leur méthode à l'analyse de journaux de navigation sur Internet (voir également les travaux de Casas-Garriga (2003) sur le sujet), Atallah et al. (2004) aux données de vente provenant de magasins.

C'est bien souvent sur la méthode de calcul de la fréquence des épisodes que diffèrent les méthodes. Alors que Mannila et al. (1997) la définissent comme le pourcentage de l'ensemble des fenêtres de taille fixe parcourant les données où un épisode est présent, Casas-Garriga (2003) basent leur mesure sur la distance maximale séparant deux événements d'un épisode (la taille de la fenêtre de recherche est alors automatiquement ajustée). Laxman et al. (2004, 2005) proposent deux mesures intéressantes⁶² : la première dénombre les épisodes sans recouvrement⁶³, la seconde dénombre les épisodes non *interfoliés*⁶⁴. Après expérimentation, Laxman et al. (2005) estiment le gain de ces mesures à un facteur 7.

Des méthodes issues de la théorie des graphes ont également été employées pour la recherche d'épisodes fréquents (Baeza-Yates, 1991; Tronicek, 2001; Hirao et al., 2001). Ces algorithmes s'avèrent néanmoins plus adaptés à la recherche et à l'extraction qu'à la découverte de tous les épisodes fréquents. Ils fonctionnent avec une phase de pré-traitement qui consiste à construire un automate fini appelé DASG pour Directed Acyclic Subsequence Graph⁶⁵. Une fois le graphe construit, un épisode peut être localisé dans la séquence en un temps linéaire (*i.e.* $O(l)$).

Les motifs avec contraintes temporelles Nous avons, dans les deux parties précédentes, présenté deux des plus importantes méthodes de recherche de motifs dans les données temporelles, à savoir la recherche de motifs séquentiels (Agrawal and Srikant, 1995) et la recherche d'épisodes fréquents (Mannila et al., 1997). Certains des travaux présentés proposaient déjà une méthode pour spécifier des contraintes temporelles : la dimension de la fenêtre glissante pour la fouille d'épisodes par exemple, ou la spécification d'un intervalle de temps maximum séparant les événements composant un épisode. Dans le cas des motifs séquentiels, Casas-Garriga (2003); Meger and Rigotti (2004) proposent une extension prenant en compte des intervalles de temps minimaux et maximaux entre les événements composant une séquence. Laxman et al.

⁶²Respectivement nommées *non-overlapping* et *non-inteleaved frequency*.

⁶³Deux occurrences d'un épisode sont dites sans recouvrement si aucune séquence d'événements appartenant à la première occurrence ne survient entre des événements appartenant à la seconde.

⁶⁴Deux occurrences d'un épisode sont dites non interfoliées si aucun événement de l'occurrence la plus tardive ne se produit avant l'événement lui succédant dans l'occurrence la plus précoce. C'est à dire h_1 et h_2 étant 2 occurrences d'un même épisode, $\forall j, 1 < j < |h_i|, h_1(v_{j+1}) < h_2(v_j)$.

⁶⁵Graphe Direct Acyclique des Sous-séquences, en français.

(2004) comme Srikanth and Agrawal (1996) proposent des adaptations de leurs méthodes respectives pour la prise en compte d'intervalles de contraintes temporelles. Une autre approche intéressante consiste à définir différents niveaux de granularité sur l'axe du temps (heure, minute, seconde, ...) et à les utiliser pour contraindre le temps entre les événements d'un motif temporel (Bettini et al., 1998).

Cependant, toutes les techniques que nous venons de voir considèrent l'évènement comme instantané. Il existe des méthodes prenant en compte la permanence dans le temps des événements. Le premier problème pour pouvoir prendre en compte cette information est sa présence dans les données étudiées. La plupart du temps, les événements se présentent sous forme d'une séquence qui provient d'enregistrement à intervalles réguliers, auquel cas la permanence d'un événement se traduit par sa répétition dans la série temporelle. Il arrive dans de rares cas que les données soient stockées sous des formes plus avancées (*c.f.* chapitre 3.3), mais la rareté de ces cas et donc des méthodes s'y appliquant oblige la plupart du temps à retransformer ces données dans une forme séquentielle brute. Laxman et al. (2002) et Lee et al. (2003) proposent néanmoins chacun des approches intéressantes pour ajouter à la description des épisodes ce type d'information de durée.

Travaux concernant la recherche de motifs

Nous allons dans la suite présenter quelques travaux intéressants dans le domaine de la recherche de motifs.

L'algorithme FTP-DS L'algorithme FTP-DS, pour Frequent Temporal Patterns of Data Streams, a été proposé par Teng et al. (2003). Il se base sur la régression pour faire de la recherche de motifs dans les flux de données. Sa particularité est de travailler sur des flux de données en alimentation permanente, avec donc une contrainte temps réel forte car les flux ne peuvent être stockés du fait de leur taille. Seul un intervalle de données – les données courantes – est donc étudié. Cet algorithme repose sur une mesure de fréquence, mais se distingue des autres méthodes par deux points :

- application de méthodes statistiques sur des flux de données permettant de résoudre les problèmes liés aux contraintes de temps,
- représentation compacte basée sur la régression permettant de résoudre les problèmes de contraintes sur l'espace et donc la taille du problème.

La difficulté du cadre applicatif de la méthode réside dans le calcul du support forcément imparfait puisqu'on ne dispose pas de toutes les données. Le balayage statistique des données permet de traiter cet écueil. Il s'agit d'utiliser une méthode simple de mise à jour du support la plus proche de la réalité. Teng et al. (2003) proposent une nouvelle définition du support, dépendant de la fenêtre et du nombre de motifs rencontrés. Ensuite, la méthode est classique, les motifs devenant fréquents quand leur support dépasse le seuil minimum. Les fréquents de taille n peuvent être utilisés pour construire des motifs fréquents de taille $n + 1$ mais seulement à l'instant d'après. Ce retard est pénalisant mais ne peut être évité quand on travaille sur des flux de données comme c'est le cas ici. Ensuite, la régression linéaire peut être appliquée car le

comptage des motifs fréquents tend vers une ligne droite. On va donc chercher la droite D qui passe au plus prêt des points via la méthode des moindres carrés.

La recherche de chroniques dans des journaux d'alarme Dousson and Duong (1999) ont développé une méthode pour l'analyse de journaux réseaux avec l'objectif de distinguer les alarmes répétitives, récurrentes ou encore régulières pour les proposer à la supervision de l'expert. Cette algorithm se rapproche fortement de l'algorithme *a priori* appliqué aux séquences d'évènements décrit par Agrawal and Srikant (1995). La particularité est que la recherche de motifs ne concerne pas systématiquement des évènements adjacents, une notion de contrainte temporelle permet en effet de trouver des règles du type « Si un évènement A survient à $t - 5$ et B à $t - 1$, un évènement C surviendra alors à $t + 2$ ».

Les algorithmes basés sur un calendrier Il existe deux types d'algorithme dans cette catégorie : ceux parcourant l'espace de recherche en largeur d'abord et ceux parcourant l'espace de recherche en profondeur d'abord (respectivement BFS et DFS selon les notations de Hipp et al. (2000)). La notion de calendrier est une représentation permettant de signaler la récurrence temporelle d'un évènement. Par exemple, $\langle 2005, *, 16 \rangle$ correspond à tous les 16^{ème} jours des mois de l'année 2005. Au niveau des algorithmes BFS, on peut citer une des nombreuses déclinaisons de l'algorithme *a priori* nommée Direct-Apriori proposé par Li et al. (2001). Sa méthode fonctionne selon trois phases. La première phase génère des ensembles de motifs candidats pour les intervalles de temps du calendrier. La deuxième phase lit les transactions sur lesquelles les dates sont couvertes par les intervalles de temps du calendrier et met à jour le support des candidats de la première phase. La troisième utilise les motifs découverts pour mettre à jour les motifs étoilés du calendrier. Pour ce qui est des méthodes de type DFS, Verma and Vyas (2005) propose un algorithme comblant les lacunes d'*a priori* dans le cas de données non denses ou très denses. Cet algorithme propose d'une part le *temporal H-mine* appliqué aux règles d'associations temporelles et d'autre part le *H-struct* et un arbre FP qui gère les ensembles de données denses.

3.4.4 La méthode des projections aléatoires

Nous allons ici nous intéresser à une méthode en particulier : celle des projections aléatoires sur laquelle nous nous sommes appuyés pour nos travaux. À partir d'une méthode proposée par Buhler and Tompa (2002), Chiu and Keogh (2003) ont proposé la méthode des projections aléatoires pour la découverte de motifs dans les séries temporelles. Cette méthode a ensuite été adaptée par puis par la suite Duchêne (2004) aux données hétérogènes et multidimensionnelles. La méthode consiste au départ à effectuer une fouille de caractères sur des données. La méthode se décompose en 3 phases : l'abstraction (ou pré-traitement) des données, la fouille de caractères, la classification et la mesure de similarité. Comme on peut déjà le constater dans cette énumération, cette méthode bien que classée dans l'extraction de motifs, conjugue en fait plusieurs techniques.

L'abstraction

Comme nous l'avons déjà présenté dans la partie 3.1, l'analyse de données réelles suppose au préalable une phase de pré-traitement de ces données afin de les rendre exploitables par les algorithmes de fouille. Dans les travaux cités concernant les projections aléatoires, nous sommes à chaque fois dans le cas d'étude de données réelles, notamment dans le cas de Duchêne (2004) qui étudie des données issues de capteurs posés sur une personne dans le cadre de la télésurveillance médicale. L'abstraction consiste donc :

- Filtrage : On supprime les hautes fréquences dans les variations des données récupérées afin de n'en conserver que les variations caractéristiques.
- Discrétisation : La discrétisation est une opération classique qui consiste à diminuer l'éventail des valeurs possibles d'un attribut. Le but est double consistant d'une part celui du filtrage de ne conserver que des valeurs caractéristiques, et d'autre part afin de diminuer la combinatoire et donc la complexité du problème.
- Agrégation : Il s'agit cette fois-ci d'intervenir sur l'axe temporel en combinant les vecteurs successifs pouvant être considérés comme similaires. On supprime ainsi les plages non-variation des données.

Fouille de caractères

La fouille de caractères consiste à identifier les sous-séquences récurrentes – les caractères – et réduire ainsi l'espace de recherche des instances effectives de motifs.

- Fréquence : Étant donné un ensemble de t séquences de taille n , on choisit de rechercher des motifs de taille l autorisant d mutations dans les instances⁶⁶. Pour cela, on choisit aléatoirement $k_{k \leq l-d}$ positions dans les l positions d'une séquence, et pour chaque séquence on calcule la sous-séquence (empreinte) qui en résulte. On répète plusieurs fois l'opération en changeant les k positions et à chaque itération, on associe à chaque projection les séquences réelles qui correspondent. De plus, on établit une matrice de collisions qui enregistre pour chaque couple de séquences le nombre de sous-séquences (*i.e.* de projections) identiques.
- Signification : Les valeurs de la matrice de collisions sont examinées selon deux critères : un seuil minimum de collisions entre sous-séquences, puis un seuil maximum de distance entre les séquences réelles correspondantes. Les sous-séquences récurrentes significatives pour l'extraction de motifs sont identifiées en trois étapes :
 - Identification d'un couple de sous-séquences satisfaisant ces deux critères,
 - Extension de ces sous-séquences pour déterminer les sous-séquences similaires « complètes », plus longues en terme du nombre de symboles qui les représente,
 - Vérification d'un seuil minimum de durée de chaque sous-séquence.
- Non redondance : Ce critère impose une démarche de synthèse des sous-séquences récurrentes identifiées d'après la matrice de collisions, afin de regrouper les sous-séquences représentatives d'une même instance de motif. Leurs représentants forment alors un en-

⁶⁶Comme l'indique le vocabulaire employé, cette technique est issue de la recherche générique.

semble de tentatives de motifs disjointes. On utilise une méthode de classification divisive.

Classification

La classification des tentatives de motifs est enfin réalisée sur la base d'une mesure de distance. On applique une méthode agglomérative non supervisée : la classification ascendante hiérarchique. Le représentant de chaque classe définit un motif.

Mesure de similarité

La mesure de similarité proposée s'appuie ainsi sur la plus longue sous-séquence commune, définie selon deux seuils de similarité sur les valeurs et dans le temps. Cette notion de similarité est étendue à la comparaison de séquences multidimensionnelles hétérogènes.

3.5 Data mining spatial

3.5.1 Les techniques de data-mining spatial

Le data-mining spatial a particulièrement été développé dans le cadre des SIG dont l'importance des contenus limite l'accès à l'information utile. Les outils de visualisation de données spatiales sont les premiers apparus pour répondre à ce problème. Aujourd'hui des méthodes basées sur les statistiques spatiales et les bases de données ont vu le jour et permettent de faire du data mining spatial. Les objectifs sont les mêmes que ceux que nous venons de présenter pour le data mining temporel : l'extraction de règles d'association du type *"si un objet est localisé au centre d'une grande ville, il se trouve à une moyenne de 15km d'une autoroute"*, le clustering permettant d'organiser les données selon leurs similarités spatiales (ex. clustering des maisons suivant leur distance d'une zone commerciale), classification permettant de trouver les caractéristiques générales d'un groupe d'entités spatiales, ainsi que la reconnaissance de formes qu'on peut également classer dans ce domaine.

L'intérêt dans notre cas serait de pouvoir extraire des règles du type *"si un individu est proche du poste de travail 1 et que le bruit émis est supérieur à 30dB alors l'individu est stressé"*. Il nous semble moins important de dégager des informations du type *"les entités proches du poste de travail 1 sont distantes du chef d'équipe d'environ 15m"*. C'est pourtant ce dernier type de règle que la plupart des techniques actuelles sont capable de trouver. L'important nous semble plus de dégager le contenu sémantique de la position d'une entité de la simulation afin que l'attribut position ne se réduise pas à un couple de coordonnées, et puisse être réellement pris en compte pour le data mining. Nous allons détailler ce point dans le paragraphe suivant.

3.5.2 Notion de sémantique d'une position et problématique

L'intégration des techniques de data mining spatial dans l'extraction de connaissances à partir de nos données de simulation nous semble encore assez incertain. En effet, la plupart de ces techniques nous paraissent peu adaptée pour l'étude de nos données de simulation. La mobilité des entités de la simulation et par conséquent les modifications fréquentes de leurs positions respectives n'est que peu prise en compte par les outils développés jusqu'alors qui concernent plutôt les données contenues par les SIG plus immobiles car le plus souvent relatives à un positionnement dans une zone géographique. Néanmoins, nous avons pu trouver dans Mennis and Liu (2003) des pistes intéressantes pour le traitement des données spatiales. Dans cette publication, les auteurs suggèrent d'appliquer des techniques de pré-traitement aux données spatiales afin d'extraire en amont du processus de data mining classique les relations spatiales entre les objets géographiques. Cela correspond en fait à transformer les données spatiales brutes en un ensemble de relations spatiales entre les différentes entités du système, relations portant sur les positions relatives des entités les unes par rapport aux autres.

C'est ce type de relations que nous nommons sémantique d'une position. Définir la sémantique d'une position consiste donc à joindre un contenu informationnel au positionnement d'une entité. Ce pré-traitement ne résout pas tout. En effet, il est nécessaire de trouver des structures permettant de stocker les relations obtenues qui lient géographiquement chaque entité à l'ensemble des autres entités du système. Mennis and Liu (2003) proposent une structure de *R-tree*, mais reste ensuite le problème de la gestion de cette structure par les processus de data mining classique ou temporel. Les recherches sur le data mining spatio-temporel étant seulement émergentes⁶⁷ et n'étant pas du tout orientées sur l'étude des données de simulation, il nous paraît difficile de statuer réellement sur les techniques à adopter pour intégrer cette dimension dans notre phase d'extraction des connaissances.

Conclusion

Ce chapitre nous a permis de présenter les principaux travaux dans le domaine de la fouille de données multidimensionnelles. La particularité d'une simulation est la production de données complexes, dimensionnellement très riches, mettant en échec la plupart des algorithmes et méthodes de fouilles de données classiques. La première difficulté est la dimension temporelle puisqu'on désire étudier l'évolution de la simulation dans le temps. Vient ensuite la dimension spatiale puisque la position des entités d'une simulation peut avoir un intérêt du point de vue de l'extraction de connaissances. Il existe déjà des méthodes de fouille de données spatio-temporelles que nous avons présentées dans ce chapitre. Nous nous intéresserons plus particulièrement à la méthode des projections aléatoires qui permet d'extraire des motifs à partir de données séquentielles. Cette méthode ne prend pas en compte l'aspect spatial des données. En sus de l'aspect spatio-temporel des données issues de simulation, le nombre d'individus ainsi

⁶⁷La première conférence sur le data mining spatial, temporel et spatio-temporel a eu lieu à Lyon en 2000 et comporte très peu d'études sur le spatio-temporel. John F. Roddick est l'éditeur des post-proceedings (Roddick, 2000).

que le nombre d'itérations d'une simulation rajoutent à la complexité du problème de fouille. Ces derniers aspects restent peu abordés dans la littérature.

Un nouveau modèle d'agent émotionnel : l'agent EFT

Ce chapitre présente le modèle d'agent émotionnel que nous avons développé et baptisé EFT pour Emotions, Feelings and Temperament afin de simuler des comportements humains crédibles. Il décrit d'une part le modèle statique de l'entité et d'autre part la modélisation des interactions entre entités spécifique au système modélisé.

Sommaire

4.1	Un modèle d'agent émotionnel : l'agent EFT	98
4.1.1	Structure générale de l'agent	98
4.1.2	Processus de contrôle de l'agent	110
4.2	Les interactions du système	114
4.2.1	Présentation du système étudié	114
4.2.2	Modélisation de protocoles d'interaction	118
4.2.3	Modélisation d'interactions spécifiques	120

Introduction

Nous avons dans les parties précédentes établi l'état de l'art sur les systèmes multi-agents, la modélisation comportementale et plus spécifiquement à la confluence de ces deux domaines les agents émotionnels. Comme nous avons pu alors le voir, si de nombreux travaux émergent dans ce domaine, peu réalise la synthèse entre agents cognitifs et agents émotionnels. D'autre part, la majeure partie de ces travaux se concentre sur l'aspect purement émotionnel au détriment de processus affectifs à l'évolution plus longue qui nous ont paru essentiels pour modéliser et simuler plus finement le comportement humain. En nous basant sur le modèle éprouvé qu'est l'agent BDI, nous avons ainsi modéliser un agent articulé autour de 3 parties : les caractéristiques personnelles, la mémoire et le système procédural. Les points précédents concernent la

modélisation de l'agent à proprement parler en tant qu'entité, nous avons également développé l'aspect modélisation des interactions entre entités avec une prise en compte de la part affective.

Nous présenterons donc dans le chapitre suivant dans un premier temps notre modèle individuel d'agent émotionnel baptisé EFT où nous détaillerons d'une part la structure statique de l'agent et l'agencement des différentes parties qui le compose, et d'autre part l'aspect dynamique de l'agent et les ressorts à la base de son comportement. Nous traiterons dans un deuxième temps de l'aspect interactionnel que l'agent entretient d'une part avec l'environnement physique dans lequel il évolue, et d'autre part avec les autres entités du système.

4.1 Un modèle d'agent émotionnel : l'agent EFT

4.1.1 Structure générale de l'agent

Notre modèle d'agent se base sur le modèle BDI auquel ont été intégrées des caractéristiques personnelles telles que les émotions ou sentiments qui vont influencer sur son raisonnement et ses choix d'actions. La division en 3 parties permet de modéliser plus finement ce qui relève des caractéristiques personnelles et constitue donc la spécificité d'un agent par rapport à un autre agent, ce qui touche à la mémoire où plus généralement au stockage des perceptions de l'agent et enfin ce qui touche au système procédural permettant à l'agent de prendre vie et d'évoluer dans son environnement. Nous présentons en figure 4.1 la structure générale de notre agent dont nous allons par la suite détailler les principales parties. Indépendamment des trois composantes que nous avons déjà citées, nous pouvons déjà voir le rôle de l'agent et les règles. Le rôle permet d'une manière générale de définir des rôles génériques que peuvent ensuite endosser différents agents. Les règles constituent la base des interactions entre le système procédural cognitif classique, structuré selon le modèle BDI, et la partie affective de notre agent. Il s'agit en fait d'une base de règles mêlant intimement les caractéristiques personnelles de l'agent à son processus de décision.

Les caractéristiques personnelles

Ce que nous avons identifié comme caractéristiques personnelles de l'agent rassemble l'ensemble des paramètres qui rendent une entité spécifique par rapport à une autre au moment de sa création à son initialisation, puis ensuite au cours de son évolution. C'est dans cette partie que nous allons donc stocker d'une part certaines caractéristiques basiques des agents telles que la position et d'autre part des caractéristiques plus spécifiques à l'agent EFT telle que son niveau de colère. Cette distinction peut paraître arbitraire car il est difficile de dissocier chez un homme ce qui a trait à la mémoire et ce qui touche à la personnalité : bien souvent la limite est ténue. Nous avons donc pris le parti de dissocier l'un et l'autre en considérant comme éléments à la base de la personnalité l'ensemble des paramètres d'initialisation et des connaissances acquises par l'agent n'étant pas utilisés dans un but procédural mais contribuant plutôt à l'orientation globale du comportement de l'agent. Pour la mémoire, nous y avons donc mis l'ensemble des

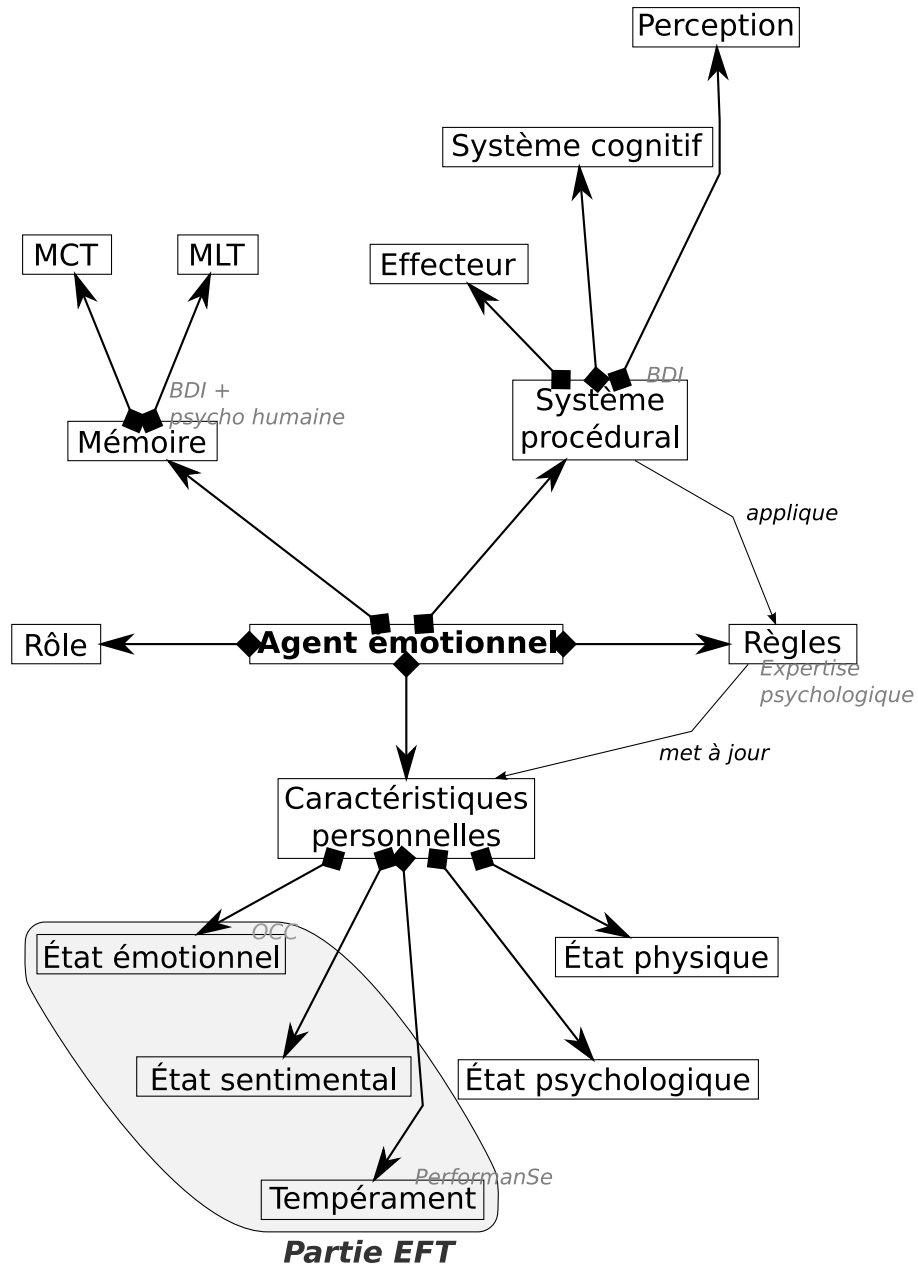


FIG. 4.1 – Modèle UML de l'agent EFT.

connaissances préalables ou acquises par l'agent lui permettant de traiter les problèmes qu'ils rencontrent et constitutifs de sa connaissance procédurale utilisée pour réaliser des plans d'action. Nous avons dissocié les caractéristiques personnelles en 5 sous-groupes : le tempérament, l'état sentimental, l'état émotionnel, l'état psychologique et l'état physique, comme le montre la figure 4.2.

Caractéristiques personnelles

Tempérament	État sentimental	État émotionnel
EXT/INT	Amitié/Inimitié	COM/JAL
ANX/DET	Confiance/Méfiance	PIT/JOM
COM/CCL	Reconnaissance/Rancune	ENG/IND
REC/DTN	Compassion/Indifférence	GRA/COL
AFF/RMC	Quiétude/Inquiétude	GOU/DEG
DIN/CIN	Estime de soi/Mésestime de soi	JOI/TRI
RIG/IMP		ESP/PEU
REA/FAC		SAT/HOR
APP/IND		SOU/DEC
POU/PRO		FRT/HON
		GRF/REM
État psychologique	État physique	
Expérience	Position (X,Y)	
Stress	Posture (debout, assis)	
Instabilité	Expression faciale (yeux, bouches, lèvres)	
Perturbation nouveauté		
Fatigue mentale		
Démotivation		
Négligence		

FIG. 4.2 – Caractéristiques personnelles de l'agent.

Émotions, sentiments et tempérament : une personnalité en trois couches Le comportement humain est fortement affecté par les émotions et l'état psychologique général de la personne. Ils sont à l'origine de réactions irrationnelles à un événement, de comportements individuels spécifiques et en partie des interactions sociales au sein d'un groupe. Les personnes traumatisées crâniens sont plus concernées du fait de leur traumatisme. Les réflexes émotionnels sont assez communément partagés entre les individus. Ne prendre en compte que cet aspect ci fait donc courir le risque d'avoir des agents dont le comportement sera peu différencié, un ensemble d'entités quasi identiques les unes aux autres dont le comportement ne variera que parce qu'elles ne seront pas forcément soumises aux même stimuli extérieurs. La personnalité d'un individu a contrario est ce qui le rend spécifique, unique dans son comportement. Elle exerce une forte influence d'une part sur ses choix rationnels et d'autre part sur la gestion des ses émotions. Ainsi, pour simuler le comportement humain, nous devons prendre en compte ces deux aspects complémentaires : les émotions et la personnalité.

C'est précisément sur la modélisation des caractéristiques personnelles d'un agent que notre modèle apporte la plus grande originalité par rapport aux différents autres modèles que l'on peut rencontrer dans la littérature. C'est d'ailleurs de cette partie qu'il tire son nom, de la version an-

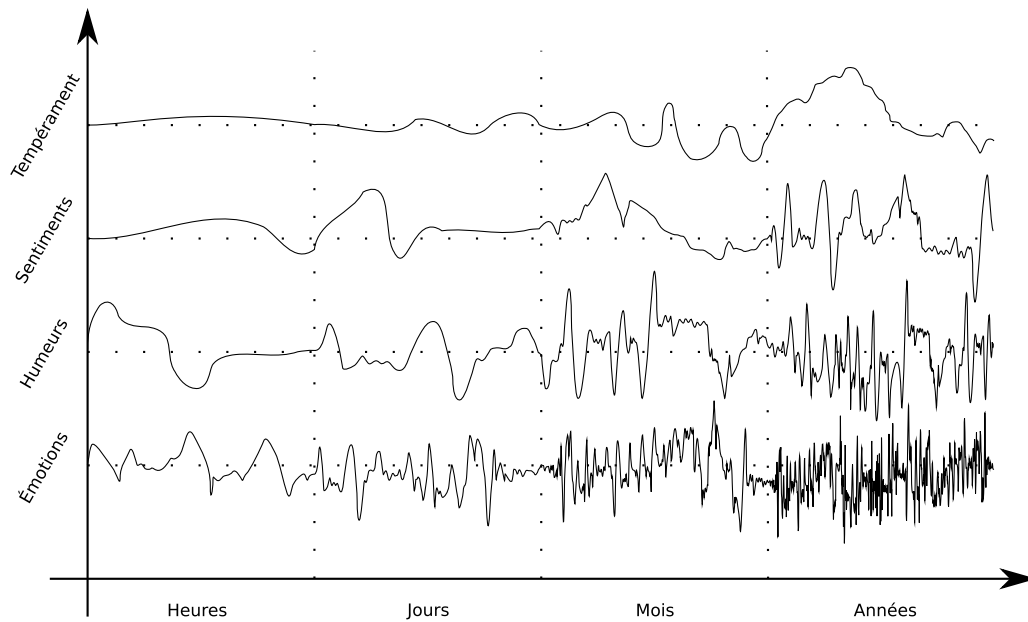


FIG. 4.3 – Variabilité des différents composants de l'affectif humain.

glaise du tryptique Émotions, Sentiments et Tempérament⁶⁸ qui sont, comme nous le verrons par la suite, les trois composants essentiels de notre modélisation de l'affect humain. En effet, nous avons à ce niveau réalisé un découpage en fonction de la permanence et de la variabilité dans le temps des différents pans qui composent la personnalité humaine (*c.f.* figure 4.3). Nous avons ainsi distingué les paramètres à forte variabilité sur de courtes périodes et faible rémanence, impliqués plutôt dans les décisions et les actions sur le vif, de ceux variant plutôt sur du long terme et ayant au contraire une forte rémanence, structurant ainsi en profondeur le comportement humain. Nous avons initialement dégagé 4 éléments différents constitutifs de la personnalité :

- les émotions : variabilité très forte de l'ordre de la minute, ne persistant pas dans le temps et ayant une influence très forte sur le comportement dans l'instant,
- l'humeur⁶⁹ : variabilité moyenne de l'ordre de la journée et influence assez forte sur le comportement sur le court terme,
- les sentiments : variabilité faible de l'ordre du mois et influence forte sur la variabilité du comportement vis-à-vis de personnes différentes,
- le tempérament (ou caractère) : variabilité quasi nulle et influence forte sur le comportement global mesuré sur le long terme.

La plupart des modèles d'agent émotionnel existants ne prennent en compte que les émotions. Nous avons considéré que cette dimension seule n'était pas suffisante. En effet, ajouté à la non spécificité des réactions émotionnelles d'une personne par rapport à une autre dont

⁶⁸EFT est l'acronyme de *Emotions, Feelings and Temperament*.

⁶⁹On rencontre quelques fois cette notion (souvent mentionné sous le vocable *mood* en anglais) dans les publications ayant trait aux émotions (Gmytrasiewicz and Lisetti, 2000).

nous venons de parler, les émotions ne traduisent également qu'un ressenti de l'instant. Leur impact est fort sur le comportement, néanmoins on ne peut considérer qu'elles déterminent à elles seules les tendances comportementales d'un individu. Le tempérament d'une part permet de traduire les tendances profondes qui caractérisent un individu par rapport à un autre. Cette dimension comportementale a également un impact sur le comportement à la fois à long terme et à court terme. Par exemple, une personne présentant un tempérament extraverti va plus facilement manifester ses émotions qu'une personne introvertie. Ces deux dimensions permettent donc conjointement de rendre un comportement cohérent et spécifique à chaque agent autant sur le court que sur le long terme. Elles ne permettent pas néanmoins de rendre compte des interactions particulières qu'un individu va développer vis-à-vis d'un autre individu. Pourtant, dans un état émotionnel identique, un individu ne va pas réagir de la même façon face à 2 personnes différentes. S'il a développé de l'amitié à l'égard de quelqu'un, il va être plus enclin à réguler sa colère.

L'humeur d'un individu joue assurément un rôle dans le comportement de l'individu, mais l'introduire dans notre modélisation aurait accru de manière fort la complexité du modèle, chacune des dimensions comportementales ayant un impact régi par des règles d'interactions. Nous avons donc considéré que, dans un premier temps, les 3 dimensions émotions, sentiments et tempérament étaient suffisantes pour modéliser un comportement cohérent.

Les modèles utilisés pour chacune des couches comportementales Pour la partie émotionnelle de l'agent, nous avons choisi le modèle OCC (*c.f.* partie 1.4.4) avec ses 11 couples d'émotions qui, comme nous l'avons vu, est particulièrement adapté à ce type de modélisation. Qui plus est, nos agents évoluent dans un univers clos dont le nombre de stimuli différents pouvant survenir est déterminé par avance. Il est donc facile dans ce contexte d'utiliser le modèle OCC en déterminant la conséquence émotionnelle de chaque événement, tâche rendue beaucoup plus difficile par un système ouvert où des événements imprédictibles peuvent survenir. Dans le modèle classique, l'intensité de l'émotion est conditionnée par 3 variables centrales. La *désirabilité* est liée aux réactions aux événements et est évaluée en fonction des buts. La *louabilité* est liée aux réactions aux actions des agents et est évaluée en fonction des valeurs morales. L'*attractivité* est liée aux réactions aux objets et est évaluée en fonction de l'état d'esprit. Nous avons pour notre part conditionné l'émergence et l'intensité des émotions avec le tempérament. Un ensemble de règles déterminent ainsi en fonction des caractéristiques comportementales d'un personne et des critères propres au modèle OCC l'émergence des émotions. Par exemple, quelqu'un qui est d'une nature plutôt extravertie et anxieuse se mettra plus facilement en colère que quelqu'un d'introverti et de calme. Nous avons également effectué une catégorisation complémentaires à celle effectuée par Ortony, Clore et Collins sur les émotions en dissociant celles n'impliquant pas autrui (honte, fierté par exemple) de celles relatives à une autre personne (pitié ou jalousie). Comme nous le verrons dans les paragraphes suivants, cela est utilisé pour dissocier les émotions influencées par les sentiments de celles qui ne le sont pas.

Pour la partie liée à la personnalité de l'individu, nous avons utilisé le modèle PerformanSe avec ses 10 dimensions comportementales bipolaires. Ce modèle présente plusieurs intérêts par rapport à nos besoins. Il est éprouvé par une utilisation à large échelle. Il a d'autre part, comme

nous avons pu le souligner dans la partie 1.2.3 été validé tant sur le plan psychologique que statistique par diverses études. La conduite de ce travail de recherche au sein de la société PerformanSe nous a également permis, d'une part, d'acquérir une bonne connaissance du modèle grâce aux experts de l'outil avec lesquels nous sommes en contact et, d'autre part, d'accéder à de nombreux profils anonymes types dans le cas d'études menées auprès d'échantillons représentatifs de personnes de certains secteurs socio-professionnels. Enfin, nous bénéficions également des études et outils connexes à l'outil d'évaluation comportemental qu'est Dialecho tel que Projequip qui nous permet de connaître les mécanismes d'attraction/répulsion entre des individus selon leur profil comportemental. Cela est indispensable d'une part pour la genèse des émotions et des sentiments et d'autre part pour les règles d'interactions entre agents comme nous le verrons dans la partie qui leur ai consacrée (c.f. 4.2).

Pour la partie liées aux sentiments, nous ne sommes pas parvenu à trouver de modèle existant satisfaisant. Ceux que nous avons pu rencontrer se réclamant d'un modèle des sentiments étaient plutôt fondés sur ce qui est considéré comme des émotions dans le modèle OCC. Il était également difficile de déterminer une base réduite de sentiments, le sujet étant éminemment subjectif, les dénominations de sentiments étaient nombreuses, conflictuelles et dont le sens linguistique qu'on pouvait leur donner prêtaient souvent à des interprétations différentes et à des recouvrements entre différents termes. Nous avons donc travaillé en collaboration avec des psychologues afin d'élaborer un modèle réduit basé sur 6 couples de sentiments en analogie avec les couples d'émotions contraires du modèle OCC et les dimensions bipolaires du modèle comportemental de PerformanSe. De la même façon que pour les émotions, ces sentiments peuvent être catégorisés en sentiments personnels : quiétude/inquiétude, estime de soi/mésestime de soi, et relatifs à autrui : amitié/inimitié, confiance/méfiance, reconnaissance/rancune, compassion/indifférence. Cette notion est intéressante parce qu'elle permet à chaque agent d'avoir sa propre personnalité (émotions et sentiments personnels), mais il peut aussi maintenir différents types de relations avec les autres agents (émotions et sentiments relatifs). Les émotions et les sentiments relatifs à une autre personne font partie de sa connaissance sociale. En effet, cela correspond à une partie de la connaissance de l'agent qui détermine ses relations sociales. À la différence des connaissances classiques du modèle BDI, celles-ci ne sont pas le résultat de la perception par l'agent de son environnement. Elles émergent également des interactions sociales avec les autres agents. Une autre différence est que cela modifie la façon dont l'agent choisit ses intentions et ses buts d'une manière non rationnelle à la différence du modèle BDI.

Des dimensions supplémentaires : les caractéristiques physiques et psychologiques En cet état, le modèle d'agent présente des lacunes, non pas au niveau de la modélisation de l'affect, mais pour tout ce qui concerne les facteurs humains à la fois physiques et psychologiques pouvant affecter sa relation aux autres et au travail. Nous avons donc, en collaboration avec des psychologues, introduit des caractéristiques supplémentaires. D'une part, 7 caractéristiques psychologiques présentées dans le tableau 4.1.1 ont été retenues. D'autre part, nous avons pris en compte 3 caractéristiques physiques permettant de traduire d'abord la position de l'agent, ensuite son expression faciale.

Expérience
Stress
Instabilité
Perturbation à la nouveauté
Fatigue mentale
Démotivation
Négligence

TAB. 4.1 – Caractéristiques psychologiques de l'agent.

Les caractéristiques psychologiques L'expérience traduit le savoir faire acquis de l'agent dans sa tâche. Outre le fait que plus celle-ci sera élevée, moins un individu commet d'erreur en effectuant sa tâche, cela a également des impacts sur un ensemble d'autres éléments : l'attention nécessaire à l'accomplissement de la tâche va être moins importante, la fatigue engendrée sera donc aussi moins grande et le stress également car l'action considérée comme plus routinière. Il serait intéressant de considérer non pas *une* expérience mais *des* expériences, autant que l'agent peut accomplir de tâches. Cela permettrait in fine d'avoir un modèle beaucoup plus réaliste. Dans notre cas, nous n'avons pas opté pour cette solution car l'agent effectue un travail précis pour lequel une expérience globale est satisfaisante.

Le stress traduit le degré de pression de l'agent. Il a un impact fort sur la qualité du travail, sur la fatigue et sur la relation aux autres (il peut être cause d'une certaine irritabilité).

L'instabilité traduit le niveau d'énervement d'un agent qui peut le conduire à avoir des réactions inattendues. Par rapport à la dimension émotionnelle, une forte instabilité entraînera une difficulté à réguler et à contrôler l'expression de ses émotions.

La perturbation à la nouveauté traduit l'adaptabilité d'un agent à des événements qui lui sont inconnus. Elle a un impact sur la gestion du changement par l'individu et peut notamment, dans le cas d'un événement imprévu, avoir une incidence sur l'augmentation du stress de la personne.

La fatigue mentale est un phénomène naturelle survenant au fil du temps. Son augmentation est conditionnée par plusieurs autres paramètres, autres les facteurs d'ambiance (luminosité trop faible, niveau sonore trop fort), le stress et le niveau d'attention nécessaire à l'accomplissement des tâches comptent pour une bonne part dans l'évolution de ce paramètre. Plus elle est élevée, plus l'attention au travail est faible.

La démotivation traduit l'état de lassitude de l'individu. Plus un individu est démotivé, plus il va être négligent et instable si son travail est générateur de stress.

La négligence traduit l'état d'attention au travail de l'individu. On parle ici d'attention au sens large (cf. partie 1.3).

L'ensemble de ces paramètres ont comme nous le verrons par la suite dans la partie concernant les règles des interactions fortes avec les paramètres affectifs : tempérament, sentiments et surtout émotions, puisqu'ils partagent avec ces dernières une forte variabilité sur le court terme.

Les caractéristiques physiques Nous avons également pris en compte dans notre modélisation un ensemble de paramètres physiques caractéristiques de l'agent. De manière classique, comme dans la plupart des simulations multi-agents où l'espace a une importance, nous avons pris en compte la position de l'agent. Dans notre simulation, un agent est assimilable à un objet mobile réductible à son centre de gravité. Chaque agent possède donc un couple de coordonnées cartésiennes (x, y) , position de son centre de gravité à un instant t dans le plan. Nous avons également eu besoin de prendre en compte la posture de l'agent (assis, debout) par rapport aux contraintes des postes de travail qu'il peut être amené à occuper. En effet, le cadre initial de notre simulation étant des individus cérébrolésés dans leur milieu professionnel, certains d'entre eux présentent des paralysies plus ou moins lourdes pouvant leur rendre insupportable une position debout. Enfin, dans le cadre de la perception de l'état émotionnel par les autres agents, ainsi qu'à l'écran par l'utilisateur humain observant la simulation, nous avons également pris en compte leur expression faciale en nous basant sur le modèle d'expression universelle proposé par Ekman et al. (1972) (*c.f.* la figure 1.5 de la partie 1.4.2). L'expression de l'agent résulte des positions de ses yeux, de sa bouche et de ses lèvres. Nous avons établi en nous basant sur les expressions universelles des primitives pour chacune de ces parties du visage. La traduction des émotions internes sur le faciès de l'agent résulte d'un ensemble de filtres (règles) où intervient notamment le tempérament (une personne extravertie montrera plus facilement et plus franchement ses émotions).

La mémoire

Comme nous avons déjà pu le voir dans la partie 1.3, il existe plusieurs types de mémoire. Dans notre modélisation, nous avons considéré la mémoire en tant que fonction cognitive de l'agent : elle permet d'organiser l'ensemble des connaissances de l'agent à un instant donné. Cette notion d'instant est essentielle, il faut en effet distinguer les connaissances stockées et utilisées sur une courte échelle de temps (le fait qu'on vienne de fermer la voiture à clés par exemple) de celles stockées et (parfois) utilisées sur une plus grande échelle de temps (le prénom de ses enfants par exemple). Nous avons fait ce distinguo dans notre modélisation en séparant mémoire à court terme (MCT) et mémoire à long terme (MLT) (*c.f.* figure 4.4).

La MCT est une structure qui va donc permettre de stocker les informations qui changent souvent ou qui sont rapidement périmées (on parlera de connaissances dynamiques). Cette mémoire là est elle-même divisée en trois sous-modules calqués sur le modèle BDI : les croyances, les désirs et les intentions. Dans notre cas, nous considérerons qu'il ne peut y avoir qu'une et une seule intention à un instant donné, il s'agit du désir de plus forte désirabilité. Cette simplification est souvent pratiquée lors de la mise en œuvre du modèle BDI, elle évite d'avoir à traiter le cas assez complexe d'intentions concurrentes qui présente peu d'intérêt pour nous.

Les croyances de l'agent sont donc les connaissances⁷⁰ qu'il a du monde qui l'entoure et de lui même. Pour ce qui le concerne, il a conscience de l'ensemble de ses caractéristiques

⁷⁰Dans le modèle BDI, on parle de croyance plutôt que de connaissance parce que rien n'assure à l'agent que ce qu'il croit est une image exacte de la réalité. Celle-ci peut être altérée par un ensemble de facteurs qui déforment la perception dont s'en fait l'agent. L'agent croit donc en quelque chose mais ne le connaît réellement à aucun moment.

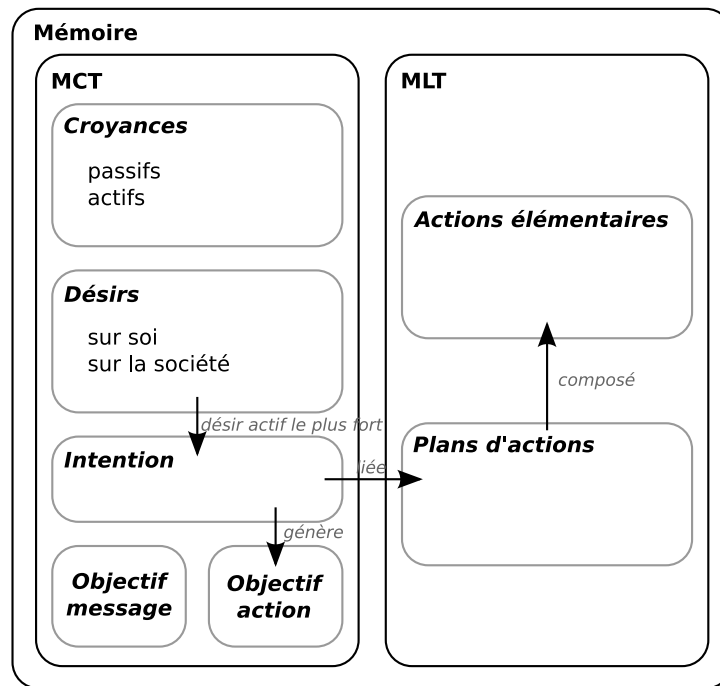


FIG. 4.4 – Structure de la mémoire.

personnelles telles que nous venons de les énumérer dans la partie précédente. C'est une vision réductrice, puisqu'en réalité l'état de colère qui peut nous habiter et la perception, et donc la connaissance, qu'on peut alors en avoir sont très deux choses bien différentes, mais cette simplification est nécessaire pour ne pas complexifier trop le modèle et suffisante dans notre cas. Pour ce qui touche à l'environnement, il va d'abord posséder des informations diverses, notamment spatiales, sur les objets et les autres entités qui l'entourent. Ensuite, il va maintenir des informations sur les rôles des autres agents ainsi que sur les relations factuelles (objectives) qu'il entretient avec eux (collaboration, subordination, etc.). Comme nous l'avons vu précédemment, les relations affectives (subjectives) qu'il entretient avec les autres agents ne sont pas maintenues dans la mémoire à proprement parler mais sous la forme des sentiments. Enfin, de la même façon qu'il possède des connaissances sur son état interne propre, il a une représentation de l'état interne perçu des autres agents (état psychologique, émotionnel, sentimental et tempérament). Bien entendu, l'image que se fait un agent de l'état interne d'un autre ne correspond pas forcément à la réalité puisqu'elle est subordonnée à la perception qu'il en a et donc à l'image que laisse apparaître l'autre agent.

Pour les désirs, nous avons choisi de les séparer en deux catégories : les désirs passifs sont les situations que l'on souhaite mais qui ne dépendent pas de nous (l'envie qu'untel nous félicite par exemple) et les désirs actifs sont les situations que l'on peut atteindre ou éviter grâce à notre action. Une intention telle que nous l'avons vue plus haut sera donc un désir actif. Afin de simplifier la mise en œuvre du modèle BDI, nous avons été contraint de scinder dans le processus de planification les actions concrètes et les actions d'expression orale. Contrairement

donc à la théorie des *Speech Acts* qui considère l'envoi de message comme des actions à part entière, nous avons opté pour deux processus différents.

La mémoire à long terme est, elle, vue comme une bibliothèque d'actions élémentaires et plans d'actions correspondant à ce que l'agent sait faire. Elle est donc peu dynamique. C'est elle qui permet à un agent de savoir ce qu'il doit faire dans une situation donnée (plans d'actions existants) ou bien de construire un plan adapté à une situation particulière (à partir des actions élémentaires existantes). Actuellement, notre agent ne prend pas en compte la planification d'actions : on considère qu'il est toujours en face de situations connues. L'objectif de notre simulation est en effet la prise en compte de l'impact de l'affect sur l'intellect, la problématique de la planification est un sujet d'étude à lui seul et est bien trop complexe pour que nous puissions l'intégrer rapidement à notre modèle avec une prise en compte cohérente des émotions dans ce processus. Pour l'instant, aucun processus ne permet donc d'enrichir la mémoire à long terme si ce n'est l'intervention humaine de l'utilisateur pour le paramétrage de l'agent. Celle-ci est donc initialisée au début de la simulation et figée par la suite. L'apprentissage de tâches n'est donc pas encore envisagé.

Le système procédural

Le système procédural correspond au fonctionnement cognitif de l'agent. Il est basé sur le modèle BDI et se compose donc d'un module de perception, d'un module de décision et d'un module d'action.

Le module de perception a pour rôle la réception des messages de l'environnement. Comme nous l'avons vu dans la partie consacrée aux ACL, ces messages peuvent être de plusieurs types. L'agent peut recevoir des stimuli d'origine environnementale (bruit, lumière, etc.), ou émis par un autre agent du système (message de type dialogue) de manière spontanée ou en réponse à un des ses messages. Les stimuli reçus passent par un filtre afin de prendre en compte les différences de perception entre individus qui peuvent être dues soit à des différences d'acuité des organes perceptifs (déficit de l'audition ou de la vue par exemple), soit, et c'est ce qui nous intéresse dans notre simulation, à un biais de perception causé par l'état émotionnel dans lequel se trouve la personne. On peut donc distinguer deux types de filtre. Tout d'abord, les messages tels que les messages de bruit ou de lumière vont être altérés ou amplifiés en fonction des capacités perceptives. Ce filtre est mécanique en ce sens qu'il ne fait intervenir ni cognition, ni émotions, mais fonctionne plus comme un filtre classique utilisé en traitement du signal qui va altérer le flux perçu. Le deuxième genre de filtre est plus intéressant car c'est celui qui va filtrer les messages de l'environnement selon les croyances de l'agent, et notamment ses émotions et ses sentiments. Par exemple, supposons qu'un agent *A* fasse une remarque malveillante à un agent *B*. Si l'agent *B* a de l'amitié pour l'agent *A*, la remarque sera mieux perçue qu'en cas d'inimitié. Ces règles ont été bâties suite à une expertise psychologique. C'est donc le message filtré qui va être transmis au module décisionnel où il sera traité.

Le module décisionnel a été construit d'une part à partir du modèle BDI et des différents travaux sur les agents cognitifs que nous avons pu énumérer dans la partie qui leur est consacrée et d'autre part sur le modèle humain que nous avons présentés en partie 1.3.5. Il est donc divisé

en deux parties : le Système Attentionnel Superviseur (SAS) et le Système de Programmation Contentive (SPC). Le premier gère les situations inconnues à l'agent. Il a donc le rôle de planificateur d'actions et de messages⁷¹. Nous avons par exemple utilisé le SAS pour modéliser et simuler la réaction d'un agent lorsqu'un imprévu survient sur la chaîne de travail. Comme nous en avons déjà parlé, à la différence d'autres simulations, le but n'est pas de construire un agent capable d'improviser de nouveaux plans d'actions à partir des connaissances dont il dispose pour résoudre les imprévus, fonction d'ordinaire dévolue au SAS. Le cœur de notre simulation concerne l'impact sur l'affect de telles situations imprévues, c'est donc ce genre de règles comportementales qui est prise en charge par le SAS. Le SPC, par contre, sert dans les situations connues. Il est composé de deux entités qui servent, pour l'une, à gérer les messages et pour l'autre, les actions. En effet, comme nous l'avons dit plus haut, messages et actions ne sont pas traités ensemble contrairement à la théorie des *Speech Acts*. Son principe général est le traitement d'un message reçu (type dialogue ou non) par la mise à jour des croyances et la génération d'une action et/ou d'un message approprié. Le module de décision sera détaillé dans la partie suivante sur le processus de contrôle de l'agent et la base de règles constituant le cœur du fonctionnement interne de l'agent.

Enfin, le module effecteur exécute l'action sélectionnée par le module de décision ou envoie le message dialogue généré. Nous aurions pu également prendre en compte sur ce module un filtre pour traduire les impacts d'une part de déficiences physiques ou mentales dans la réalisation de certaines actions ou l'élocution des messages verbaux, d'autre part de l'influence de l'état émotionnel sur les actes. En effet, dans certains cas, la pression émotionnelle peut être telle par exemple qu'un individu est incapable de contrôler ses paroles et ses actes. Nous avons choisi de traiter ce genre de cas en amont dans le SPC via la base de règles en considérant que le choix d'une action ou d'une parole émane d'un processus cognitif impliquant les émotions.

La base de règles

Le dernier composant de notre agent est sa base de règles. Cette partie est très importante car c'est elle qui contient toute l'expertise psychologique formalisée en terme de « *Si . . . , alors . . .* ». Elle contient le modèle de fonctionnement d'un agent. Cette base est construite comme une bibliothèque organisée de règles affectant l'ensemble des processus perceptifs et décisionnels de l'agent. Chaque règle est naturellement structurée en deux parties classiques : la prémisse et la conclusion. Ensuite, cinq grandes entités entrent en jeu dans nos règles :

- les caractéristiques personnelles de l'agent,
- les croyances qu'il a sur l'environnement,
- les croyances qu'il a sur les autres agents,
- la présence d'un message (dialogue ou information),
- la présence d'un objectif à accomplir ou accompli.

⁷¹Dans cette partie, nous distinguons messages et actions pour faciliter la compréhension. Il est à noter que la communication entre agents se base sur l'envoi de messages. En pratique, une action sera, en fait, l'envoi d'un message particulier.

Pour chaque famille de règles, nous avons tenté de déterminer les entités mises en jeu, ce que montre le tableau 4.2.

TAB. 4.2 – Formalisation des familles de règles (familles de 1 à 10).

	Prémisse					Conclusion				
	Caractéristiques Personnelles	Croyance Environ- nement	Croyance Agent	Message	Objectif	Caractéristiques Personnelles	Croyance Environ- nement	Croyance Agent	Message	Objectif
1	✓		✓	✓					✓	
2	✓	✓		✓					✓	
3				✓		✓		✓		
4	✓	✓		✓		✓	✓			
5	✓				✓	✓				
6	✓	✓	✓						✓	
7	✓		✓						✓	
8	✓		✓	✓					✓	
9	✓	✓								✓
10	✓					✓				

Nous allons maintenant présenter les familles de règles que nous utilisons et fournir, pour chacune, un exemple exprimé en langue naturelle. Les règles présentées en exemple sont volontairement plus simples que celles réellement mise en œuvre dans la plate-forme afin d'en faciliter la compréhension.

La classe 1 est constituée des règles pour filtrer les messages de type dialogue. Nous avons déjà présenté l'intérêt du filtre dans la partie sur le système procédural. Le filtre prend en compte le message, la personne émettrice et l'état de l'agent à l'instant de la perception et altère légèrement le message pour prendre en compte les relations affective entretenues avec l'entité émettrice du message et l'état émotionnel de l'agent récepteur au moment de la réception. Nous pouvons par exemple avoir une règle du type : « ***Si** un agent A me fait une remarque avec une intention bienveillante (message), si je suis en amitié, en confiance avec lui et si je ressens de la gratitude envers lui (caractéristiques personnelles) et si je crois qu'il est en amitié avec moi (croyances sur l'agent A) **alors** je perçois la remarque avec une intention très bienveillante (message modifié)* ».

Nous regroupons dans la famille 2 les filtres des messages environnementaux. Nous considérons que ce sont des filtres mécaniques au sens où nous l'avons entendu ci-dessus. Ils ne sont donc pas liés à une expertise psychologique mais plutôt à des phénomènes physiques (défiance auditive, visuelle, etc.). Par exemple, « ***Si** je possède une surdité légère (caractéristique*

physique) et que je reçois un message environnemental bruit d'ambiance de 70dB alors je le perçois avec 20% atténuation. »

Les groupes 3, 4 et 5 concernent les mises à jour des différents composants internes de l'agent (mémoire, but, émotions, sentiments). En 3, nous mettons les règles portant sur la mise à jour suite à la réception d'un message de type dialogue. En poursuivant l'exemple de la première classe de règles, on pourra donc avoir des règles du type : « *Si j'ai reçu de l'agent A une remarque très bienveillante alors ma joie, mon amitié pour lui et ma quiétude augmentent et je renforce ma croyance en son amitié pour moi* ». On considère, pour simplifier, que les caractéristiques personnelles de l'agent ne sont pas prises en compte dans la prémisse puisqu'elles jouent leur rôle dans le filtrage du message.

Ensuite, la partie 4 concerne les mises à jour suite à un message environnemental. C'est à partir de ces règles qu'un agent va mettre à jour ses connaissances sur le monde extérieur (position des objets qui l'entourent, luminosité de la pièce, etc.). Dans le cadre du travail sur une chaîne de production, on pourra donc avoir par exemple une mise à jour du nombre d'éléments à traiter.

La famille 5 concerne les mises à jour des caractéristiques personnelles après l'accomplissement d'un objectif. Nous avons, par exemple, « *Si j'ai réussi à faire cette action que je jugeais difficile, alors ma fierté augmente* ». Cette règle est tirée du modèle OCC.

Les règles de types 6, 7, 8 et 9 servent à générer des messages de communication ou des actions. Pour une meilleure formalisation des règles, nous avons séparé la génération des messages dialogue en trois sous-groupes. La famille 6 concerne les messages pour le travail. Le message typique est l'appel du chef de ligne pour un non fonctionnement de la machine. Dans la famille 7, nous avons regroupé les règles de génération d'un message relationnel du type encouragement à un collègue si je perçois qu'il fait bien son travail, en 8 la génération des messages en réponses à une question qui m'est posée et en 9, les règles permettant de générer des objectifs actions. Ces quatre familles de règles portent plus sur les interactions entre agents que sur l'agent lui-même, nous ne les détaillerons donc pas dans cette partie mais plus spécifiquement dans la partie consacrée aux interactions (c.f. partie 4.2).

Enfin, la famille 10 concerne l'évolution interne de l'agent. En effet, ses caractéristiques personnelles sont étroitement liées et on imagine mal qu'un paramètre puisse évoluer sans affecter les autres. Nous aurons donc par exemple des règles du type : « *L'anxiété influe sur le stress* », « *Si la perturbation à la nouveauté baisse, alors le stress baisse* ».

4.1.2 Processus de contrôle de l'agent

L'ensemble des composants de notre modèle d'agent restent inutiles si nous ne présentons pas les processus par lesquels un agent s'anime et qui interconnecte ces différents composants. Nous allons donc dans la partie suivante détailler le processus dynamique qui lui permet d'agir en réaction à un événement, d'engager une discussion ou encore de répondre à quelqu'un. Nous verrons dans un premier temps, le fonctionnement général de notre agent puis nous insisterons sur la place des caractéristiques personnelles de l'agent dans son raisonnement.

Le fonctionnement de l'agent

Le processus de contrôle de notre agent est basé sur le fonctionnement d'un agent BDI tel que nous l'avons présenté en bibliographie (c.f. partie 2.2.3). Il est représenté en figure 4.5.

Le premier choix que nous avons fait a été de gérer la communication et l'action séparément dans un souci de simplification. Toutefois, ces deux processus fonctionnent en parallèle, ce qui permet à l'agent de pouvoir à la fois parler et agir. Au niveau du traitement des conversations, tous les messages que reçoit l'agent sont stockés dans un tampon, ce qui permet à l'agent de les traiter un par un, dans leur ordre d'arrivée (FIFO). C'est un système classique dans les systèmes multi-agents communément appelé *boîte aux lettres*. Néanmoins, le fonctionnement FIFO peut être remis en cause dans le cas de messages urgents reçus par l'agent (une alerte au feu par exemple va interrompre la conversation courante). Pour gérer ce type de cas, un champ explicite dans les messages échangés indique la priorité du message. L'agent reçoit donc un message que nous qualifions de non intentionnel car il n'a pas fait de requête pour que quelqu'un lui parle. Son envie de dialogue est stockée dans ses désirs passifs, elle ne dépend donc pas de lui. Le message reçu est filtré par les règles de type 1 que nous avons vues plus haut. Ensuite, l'agent met à jour ses croyances sur l'agent émetteur et ses caractéristiques personnelles (règles 3). Enfin, si une des règles 8 est vérifiée, l'agent génère un objectif message, en réponse au message reçu, que le module effecteur va s'occuper d'envoyer. Il effectue cette succession d'actions tant qu'il a des messages à traiter dans son tampon de messages conversationnels. S'il n'en a pas, il surveille régulièrement l'arrivée de nouveaux messages.

Le schéma montre que le processus de gestion des messages conversationnels n'est en fait pas complètement séparé du processus de contrôle de l'action. En effet, la réception d'un message entraîne un changement des caractéristiques personnelles et nous avons vu que les règles de la classe 9 permettent de générer des objectifs actions en fonction de son état personnel. Nous allons maintenant présenter le fonctionnement de notre agent pour gérer ses désirs et ses actions. Comme pour les messages conversationnels, l'agent stocke tous les messages non conversationnels de son environnement dans un tampon pour les traiter un à un. On entend par message non conversationnel un message émis « par » l'environnement à l'agent. Ces messages ne sont en fait pour la plupart pas générés inopinément par l'environnement mais sont perçus par une démarche intentionnelle de l'agent qui scrute son environnement pour mettre à jour ses croyances. Par exemple, dans le cas que nous allons étudier où un agent opère sur une chaîne de production, pour rester sur notre exemple, l'agent a besoin de connaître l'état de la machine pour pouvoir agir dessus et va donc intentionnellement consulter cet état. C'est pourquoi on parle dans le processus de contrôle de perception intentionnelle. On peut dans certains cas recevoir des messages inopinés, le déclenchement d'une alarme par exemple.

Ensuite, l'agent met à jour ses croyances en fonction de l'ensemble des stimuli perçus et filtrés. Puis, il modifie son objectif à partir de son intention courante. En effet, une intention est le désir de plus forte désirabilité auquel est associé un plan d'actions élémentaires. L'objectif est donc automatiquement l'action élémentaire suivante dans le plan d'actions. Faire cette mise à jour de l'objectif courant après l'actualisation des croyances a pour intérêt la prise en compte par l'agent des changements qu'il a pu faire sur l'environnement lors de son action précédente.

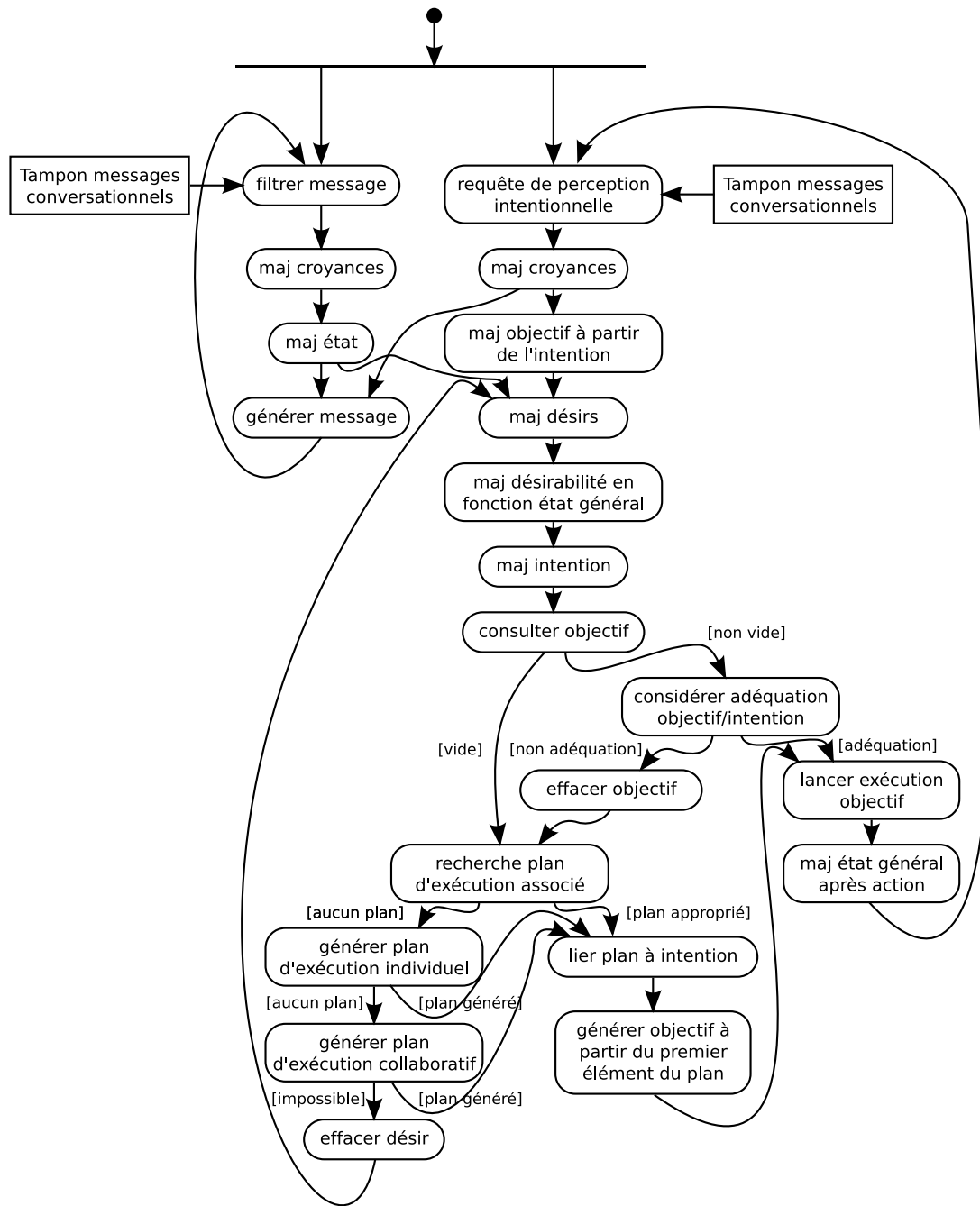


FIG. 4.5 – Processus de contrôle de l'agent.

Il va ainsi au choix soit poursuivre son intention originelle suite au bon déroulement des étapes précédentes du plan d'actions (les actions qu'il a entreprise sur l'environnement l'ont fait passé dans l'état voulu), soit abandonné son intention pour une autre ces actions n'ayant pas permis de modifier l'environnement de la manière désirée. Ensuite, l'agent va mettre à jour sa liste de désirs grâce aux règles 9 et par la suite changer les désirabilités associées. Une fois les désirs à jour, l'agent va sélectionner le désir le plus fort et le transformer en son intention. Cette phase est très intéressante car elle permet de changer d'intention même en cours d'exécution (le changement s'effectue entre deux actions élémentaires du plan de réalisation). Ceci fait que notre agent fonctionne en obligation ouverte et est donc très réactif aux changements de l'environnement.

Après cette phase, si l'objectif de l'agent est en adéquation avec son intention alors, il l'exécute et met à jour son état général avec les règles de la famille 5 et il recommence le processus. Par contre, s'il n'a pas d'objectif ou si son objectif n'est plus lié à son intention (réactivité à l'environnement), il va chercher le plan d'actions associé à sa nouvelle intention (SPC) ou bien générer un nouveau plan pour réaliser cette intention (SAS). Une fois ce plan établi, l'agent génère comme objectif courant le premier élément du plan et l'exécute. Le processus recommence alors en mettant en objectif le second élément du plan.

Cette description du processus de contrôle montre que notre agent a bien un fonctionnement basé sur la réalisation de son intention qui est fonction de ses désirs et de ses croyances. De plus, c'est un agent BDI à obligation ouverte puisque la réalisation d'une intention peut être interrompue. Nous proposons donc un agent capable d'agir et de réagir en fonction de l'environnement ce qui est primordial pour notre simulation où l'environnement est assez dynamique. Nous allons maintenant insister plus particulièrement sur l'intervention des émotions, sentiments et tempérament dans ce processus.

La place des émotions au sein du processus de contrôle

Les émotions, et plus généralement de l'état interne de l'agent, jouent un rôle primordial dans les processus cognitifs. Tout d'abord, nous avons vu dans l'état de l'art que Damasio (1994) avait prouvé que sans émotion, aucune prise de décision et aucun raisonnement n'étaient possibles. Bien entendu, autant cette règle s'avère applicable pour un Homme, autant elle est caduque dans le monde informatique où l'ensemble des actes réalisables et réalisés d'un programme sont pré-programmés et où donc l'émotion ne peut être qu'un paramètre de plus en aucun cas indispensable pour la prise de décision. Néanmoins, pour intégrer cette notion de neuropsychologie, nous avons choisi de mettre des variables émotionnelles en pré-condition d'une action. Ainsi, suivant l'état de l'agent, il ne peut pas choisir de faire toutes les actions qu'il connaît. Par exemple, en prenant le contexte de la chaîne de production, s'il est en colère, il va jeter les paquets du tapis plutôt que les enlever normalement. L'émotion n'est donc pas à la base de la prise de décision puisque sans elle, il accomplirait sa tâche normalement, mais c'est un paramètre qui peut faire basculer son choix. Ensuite, nous avons introduit la notion de sentiments pour gérer les relations humaines et les relations de travail. Ils influencent aussi l'action et la communication dans la mesure dans laquelle nous l'avons déjà décrit. Enfin, le

tempérament permet de faire varier les émotions et les sentiments d'une personne différemment. En effet, une personne anxieuse sera plus sensible à l'émotion de peur qu'une autre.

Après avoir pris en compte l'influence cognitive des émotions, nous avons aussi essayé de considérer l'aspect physiologique de celles-ci. C'est pourquoi, notre agent a une expression faciale permettant aux autres agents de deviner ou supposer son état émotionnel. Dans un premier temps, nous avons limité l'expression des émotions à la théorie d'Eckman. Mais grâce à la présence d'un état physique dans notre agent, il sera assez simple d'introduire de nouveaux indices émotionnels tels que le timbre de voix, les épaules ou les bras.

On voit donc que l'affect de l'agent a un impact fort sur l'ensemble de ses actes, autant au niveau du raisonnement que du côté physique. De plus, l'architecture de notre modèle basée sur des composants interagissant à travers une base de règles peut aisément être enrichie en ajoutant simplement de nouvelles règles aux typologies déjà existantes ou en créant de nouvelles typologies. On pourrait par exemple imaginer que l'état émotionnel de l'agent puisse avoir une influence sur ses caractéristiques physiques. Un agent heureux sera sans doute plus énergique et donc moins sensible à la fatigue. Néanmoins, ce genre de règle doit être validé par une expertise psychologique avant d'enrichir le modèle sous peine de simuler des comportements incohérents.

4.2 Les interactions du système

4.2.1 Présentation du système étudié

Description du système

Le système étudié est une chaîne de re-conditionnement de lots. Sur cette chaîne représentée sur la figure 4.6, les employés déconditionnent des lots qui leur arrivent sur palette. Puis, ces lots sont déposés sur un tapis roulant, acheminés jusqu'à une filmeuse et un four de rétraction du film. En sortie, une étiquette est déposée sur le lot conditionné. Le lot est ensuite emballé et le carton une fois plein passe dans une scotcheuse avant d'être palettisé. Dans le prototype de faisabilité réalisé en 2003, nous nous étions intéressés aux postes 6, 7, 9, 10 et 11. Nous avons réduit notre champ de représentation aux postes 6, 7, 9 et 10 afin, d'une part, de partir d'un sous-exemple pour le complexifier ensuite comme le préconise l'approche systémique selon Méléze (1972), et d'autre part, de faciliter la validation et la correction de notre modèle.

Identification des entités et de leur structure relationnelle

Identification des entités Premièrement, le poste de travail constituera une entité composée de sous-entités : four, tapis de sortie de four, tapis d'acheminement des cartons. Ces entités sont modélisables par des agents purement réactifs. Le comportement de la machine étant relativement simple, l'architecture d'agent réactif correspond bien au mode de fonctionnement à modéliser : la machine reçoit un stimulus (tel que l'actionnement du bouton de mise en marche) et réagit (ici, se met donc en marche). On parlera d'agent machine. Ensuite, les lots et les car-

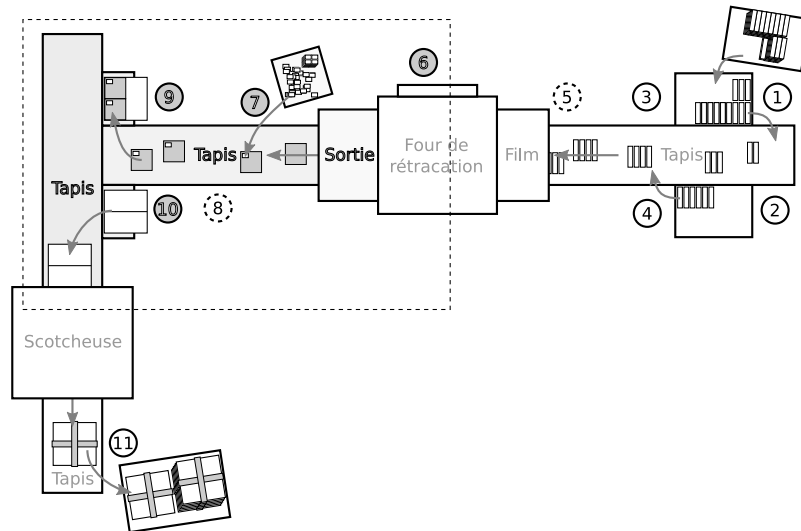


FIG. 4.6 – Chaîne de conditionnement.

tons peuvent être considérés comme des entités inertes (ou objets) du système pouvant être manipulés par les agents du système.

Enfin, chaque individu travaillant sur la chaîne constituera une entité. Cette entité sera modélisée par un agent cognitif émotionnel. Comme nous l'avons vu dans la partie 2.2.3, ce type d'agent est en effet particulièrement adapté à la modélisation des comportements humains. On parlera alors indifféremment d'agent humain, émotionnel ou d'avatar. Parmi ces agents, seul l'agent 6, le chef de ligne, n'est pas traumatisé crânien. En tant qu'encadrant, il a la charge de s'assurer du bon fonctionnement de la chaîne et de remédier aux problèmes qui peuvent survenir. Notre agent est constitué d'une mémoire (correspondant aux croyances d'un agent BDI), de caractéristiques personnelles (émotions, sentiments, tempérament, état psychologique, état physique) et d'un système délibératif basé sur une architecture BDI avec une prise en compte des caractéristiques personnelles.

Nous considérerons une dernière entité appelée environnement qui ne possède pas de réelle existence, mais va permettre de gérer l'ensemble des phénomènes ne dépendant pas des entités déjà citées. On peut donner l'exemple de la diffusion de la lumière ou des lois physiques qui vont réduire la zone d'audition d'un agent ou sa visibilité. L'environnement prend en fait le rôle d'administrateur du système. C'est grâce à lui que l'on va définir l'ensemble des lois, externes aux agents, qui régissent le système.

Structure relationnelle L'ensemble des personnes travaillant sur la chaîne entretiennent des relations de dépendance. Nous allons utiliser la notation matricielle présentée en 1.7.2 pour caractériser ces dépendances afin d'identifier les différentes interactions entre agents. Nous avons préféré distinguer les interactions entre agents traumatisés et entre un agent traumatisé et le chef de ligne, l'agent, non traumatisé. En effet, les agents 7, 9 et 10 travaillent sur la chaîne en

$$\begin{array}{c} 7 \quad 9 \\ 10 \left(\begin{array}{cc} \bar{a}id_{7,10}\bar{g} & \bar{a}id^*g \\ \bar{a}id_{7,9}\bar{g} & \end{array} \right) \quad 6 \left(\begin{array}{c} \# \\ \bar{a}id_{6,\#}g_{6,\#} \end{array} \right) \end{array}$$

FIG. 4.7 – Structures relationnelles entre agents.

continu, alors que le chef de ligne ne fait qu'intervenir en cas de problème.

Ces matrices nous indiquent que les agents 7 et 9/10 travaillent séparément sur des tâches différentes mais dépendant l'une de l'autre. Le fait que les agents 9 et 10 soient dépendant d'un autre agent plutôt que d'une machine est intéressant car cela permet d'envisager des interactions sociales entre ces agents. Les agents 9 et 10 ont des tâches interdépendantes et échangeables. Encore une fois, cela induit des interactions sociales entre ces deux agents pour synchroniser leurs tâches et s'aider si besoin est. À tout moment, en cas de problème, le chef de ligne peut intervenir pour aider localement l'agent ou remédier à un problème de machine. Le choix de ces agents est pertinent car il permet d'observer plusieurs structures relationnelles dans un même groupe de travail. D'un point de vue de la simulation, le contenu informationnel pouvant être dégagé est donc plus riche. La synthèse des structures relationnelles entre agents est présentée en figure 4.7.

Diagramme de cas d'utilisation

Dans un premier temps, nous allons utiliser une modélisation purement UML. Nous proposons d'utiliser les diagrammes de cas d'utilisation pour fournir une vue globale des différentes interactions qui peuvent exister entre les agents du système⁷². Après recueil de l'expertise auprès des personnes en contact avec le milieu que nous devons modéliser, nous avons dégagé un certain nombre d'interactions entre individus observables au sein de l'atelier. Avec le même souci de ne conserver qu'un ensemble restreint d'exemples illustratifs qui nous a menés à sélectionner un sous-ensemble d'agents, nous avons choisi un certain nombre d'interactions caractéristiques du milieu modélisé et à l'origine de la dynamique dans les rapports sociaux. Il existe des SMA privilégiant le langage, ce n'est pas notre cas. Les interactions sélectionnées sont simples et à but illustratif et privilégient donc l'aspect réactif de l'agent à l'aspect dialogue. Le diagramme de cas d'utilisation présenté sur la figure 4.8 présente ces différentes interactions d'un point de vue global. Nous avons distingué les interactions qui consistent à communiquer par la parole avec autrui, à apparaître à autrui, à agir sur l'environnement de manière physique et à percevoir son environnement de manière volontaire (regarder une entité de l'environnement et observer son état apparent) et involontaire (capter le bruit émis par la machine).

Au niveau de chacun des postes, nous avons ensuite mené la même démarche mais cette fois-ci uniquement pour les actions de l'avatar sur la machine, les autres interactions n'étant pas spécifiques à un poste donné. La figure 4.9 présente le diagramme établi pour le poste 9.

⁷²Nous n'utiliserons pas ici ces diagrammes comme on peut le faire en génie logiciel pour caractériser les cas d'utilisation entre utilisateur et plate-forme de simulation.

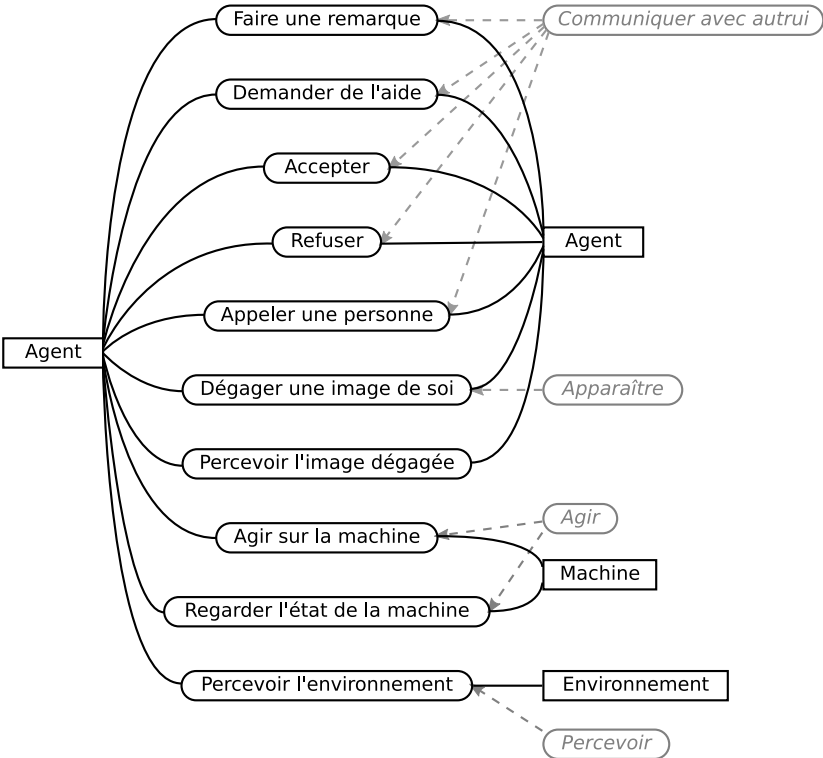


FIG. 4.8 – Diagramme de cas d'utilisation global

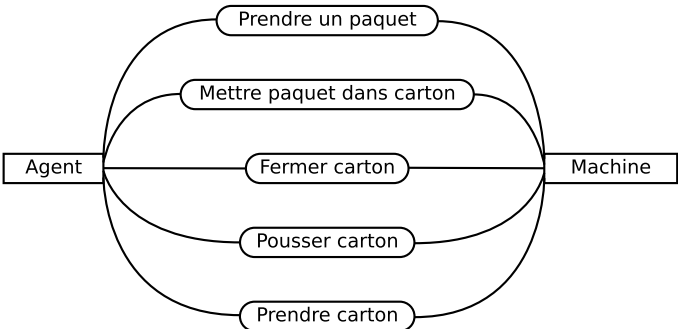


FIG. 4.9 – Diagramme de cas d'utilisation du poste 9.

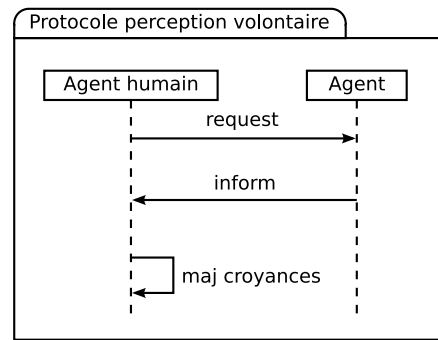


FIG. 4.10 – Protocole de perception volontaire.

4.2.2 Modélisation de protocoles d'interaction

Protocole de perception volontaire

La perception volontaire correspond pour nous à l'observation par un agent d'un autre agent (individu ou machine) du système. La figure 4.10 présente le protocole de perception volontaire de l'environnement. Afin de conserver le mécanisme d'envoi de message entre agent, nous avons convenu que cette consultation se faisait en deux étapes : un message requête à la machine pour qu'elle retourne un message d'information. Nous aurions pu opter pour une consultation directe par l'agent humain des paramètres de la machine, mais cela comporte des désavantages. D'une part, l'état visible peut être différent de l'état interne, le message envoyé par la machine ne doit donc véhiculer que ce qui est visible. D'autre part, cela ne correspond pas au fonctionnement agent classique, ce qui peut nuire au modèle et à sa portabilité sur différentes plate-formes de conception. Connaître l'état de la machine par des accesseurs fait passer la machine du statut d'agent à celui de simple objet.

Protocole d'interaction avec le poste de travail

Le diagramme de la figure 4.11 présente le protocole général d'interaction d'un individu avec la machine, que nous avons nommé protocole travail. Dans un premier temps, l'employé perçoit intentionnellement la machine. Nous utilisons une notation d'AgentUML pour faire référence au protocole de perception intentionnelle défini plus haut.

Dans un second temps, l'agent met à jour ses caractéristiques personnelles ainsi que ses désirs. Cela correspond au processus délibératif interne à l'agent décrit dans la partie 4.1.2.

Enfin, l'agent effectue une action sur la machine qui se met à jour en conséquence. Il est important de noter que la machine est un agent purement réactif. Elle n'a pas de réelle autonomie dans son comportement face à un message. Sa réaction est définie par des règles bien précises fonction de son état actuel (si la machine est en marche, qu'elle reçoit un message d'arrêt, elle s'arrête). Or, nous avons voulu nous conformer aux performatives définies par le langage FIPA-ACL pour les raisons décrites en partie 2.3.4. Ce qui fait que l'action se traduit

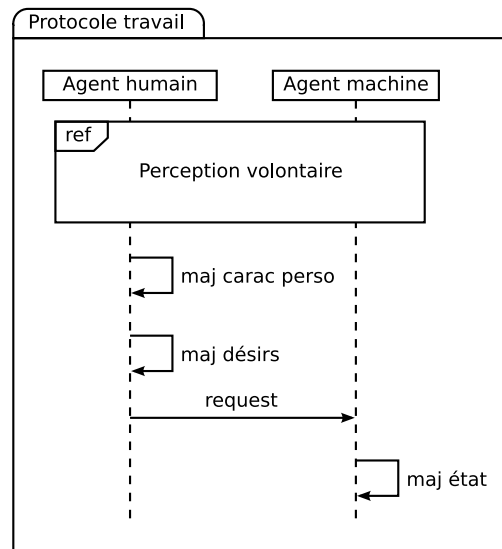


FIG. 4.11 – Protocole d'interaction avec le poste de travail.

par un message *request*. On ne peut donc faire figurer sur le diagramme la distinction entre les messages de perception et les actions, d'où un certain manque d'expressivité.

Les diagrammes d'interaction entre individus

Les diagrammes des figures 4.13 et 4.14 représentent respectivement les protocoles remarque et demande d'aide entre individus. Ce sont les deux comportements entre individus traumatisés les plus fréquents que nous avons pu rencontrer sur la chaîne de travail. La remarque a lieu quand une personne travaillant sur la chaîne constate qu'une tâche s'est mal déroulée en amont de son poste (mauvais étiquetage, absence de paquet, etc.). La demande d'aide est formulée quand la personne ne parvient pas à réaliser seule une tâche et se retrouve dans une situation qui peut devenir bloquante (trop de paquets sur le tapis, plus d'étiquettes à apposer, etc.). Afin de traduire l'impact des émotions sur la communication entre individus, nous avons introduit un coefficient d'intention dans chaque échange de paroles qui traduit son degré d'amabilité (ou de bienveillance). Ce coefficient se voit attribuer une valeur entre 0 et 1 (figure 4.12). D'autre part, dans le cas de la remarque, nous avons considéré qu'une remarque pouvait être positive ou négative. Une remarque positive correspond à une félicitation⁷³, une remarque négative à un reproche.

Dans le cas d'une remarque, le message se traduit par un *inform*. À la réception de ce message, l'ensemble des caractéristiques de l'agent vont être mises à jour par des règles internes. L'évolution de cet état va conditionner le fait que le destinataire réponde ou non. Nous avons utilisé la notation AUML *option* pour distinguer ces deux choix. Dans le cas d'une réponse

⁷³D'après les experts, ce sont plus des reproches que des félicitations qui circulent sur la chaîne que nous modélisons. Néanmoins, pour l'extensibilité du modèle, nous avons voulu prendre les deux en compte.

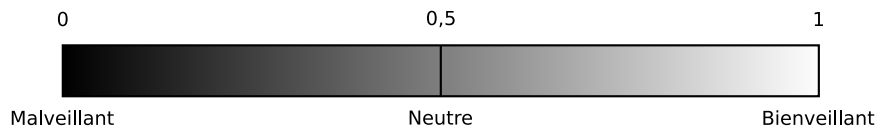


FIG. 4.12 – Coefficient d'intention d'un message dialogue.

par un nouvel *inform*, l'intention du message va être établie par des règles prenant en compte l'état interne de l'agent. Dans certains cas, une remarque peut amener le destinataire à changer de comportement en fonction de la remarque, c'est à dire à générer de nouveaux désirs. Nous n'avons pas donné la possibilité à nos agents d'échanger plus de messages. D'une part, cela correspond encore une fois à une réalité du système étudié. D'autre part, cela permet de privilégier des scénarii dynamiques où les agents vont échanger peu mais souvent, le but n'étant encore une fois pas d'axer la simulation sur le processus conversationnel mais plus sur l'impact des interactions entre agents sur leur état interne (particulièrement émotionnel).

La demande d'aide se traduit par un message de type *request* où l'agent initiateur va préciser la tâche qu'il souhaiterait voir accomplie par le récepteur. La réception d'un tel message va entraîner une mise à jour de l'état interne de l'agent récepteur. Suivant certaines règles qui vont prendre en compte ses compétences pour réaliser la tâche demandée et ses caractéristiques personnelles, émotionnelles et sentimentales vis-à-vis de l'agent initiateur, il va donner son accord (*agree*) ou non (*refuse*) pour réaliser la tâche. Nous tenons à insister sur le fait que la demande d'aide n'entraîne pas de processus de négociation entre les deux agents comme on peut en trouver dans de nombreux SMA collaboratifs. Ce type de comportement n'est pas caractéristique de ceux que l'on peut rencontrer sur une chaîne de travail. Cette interaction a donc plus été considérée dans le but de générer une réaction émotionnelle du type : « Je suis déçu car untel a refusé de m'aider. » ou « Je suis reconnaissant envers untel pour son aide. », que dans celui d'engendrer un dialogue entre individus.

4.2.3 Modélisation d'interactions spécifiques

Interactions avec la chaîne de production

Une description précise du fonctionnement de la chaîne de production est nécessaire pour identifier l'ensemble des situations auxquelles peut être confronté chaque avatar. Pour cela, nous avons d'abord établi un diagramme d'activités correspondant au fonctionnement normal de la machine (figure 4.15). Par normal, on entend que chaque individu travaille correctement sur la machine. Ce diagramme fournit une vue globale non détaillée du processus de travail sur la chaîne. À chaque agent, correspond un couloir d'activité⁷⁴. On peut observer l'enchaînement des tâches sur la machine, les rôles de chacun des avatars ainsi que les flux d'objets entre agents (flèches pointillées sur le dessin). L'ensemble des dépendances relationnelles précédemment caractérisées apparaît sur le diagramme (dépendance par rapport à un flux d'objet ou par rap-

⁷⁴Dans celui de la machine, une barre de synchronisation horizontale permet de modéliser l'itération des paquets sortant du four, cela pouvant conduire à un état de bourrage où intervient l'avatar 6.

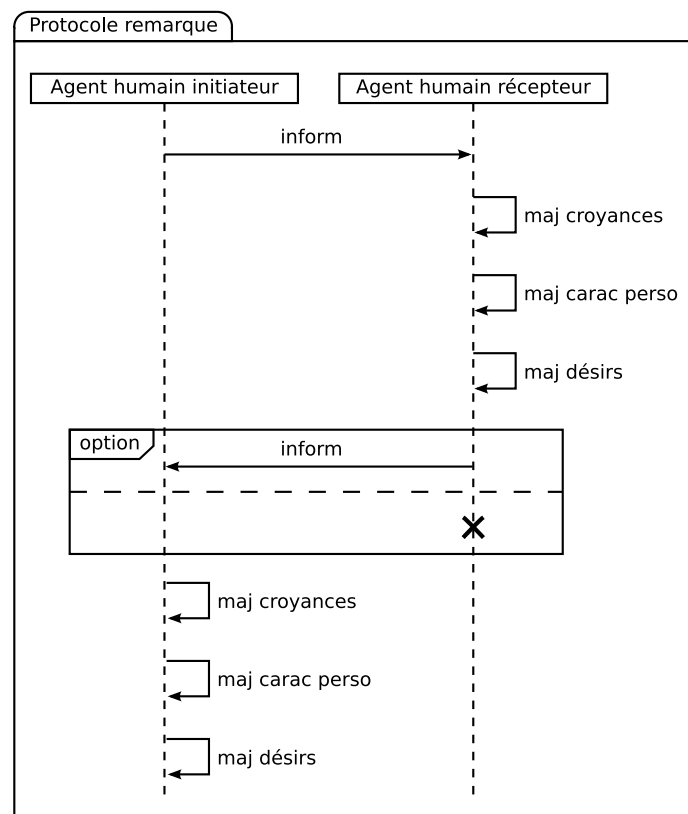


FIG. 4.13 – Protocole de remarque entre individus.

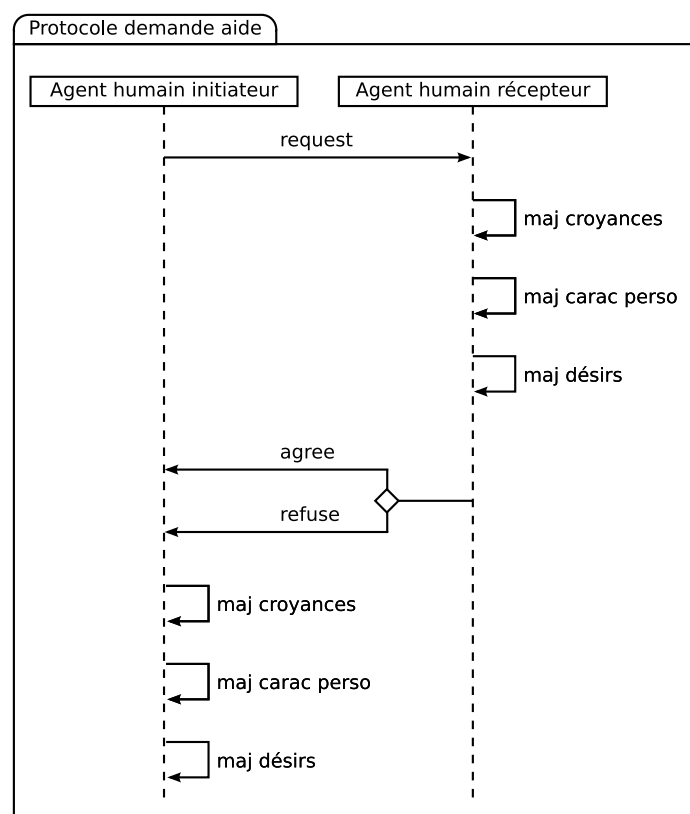


FIG. 4.14 – Protocole de demande d'aide entre individus.

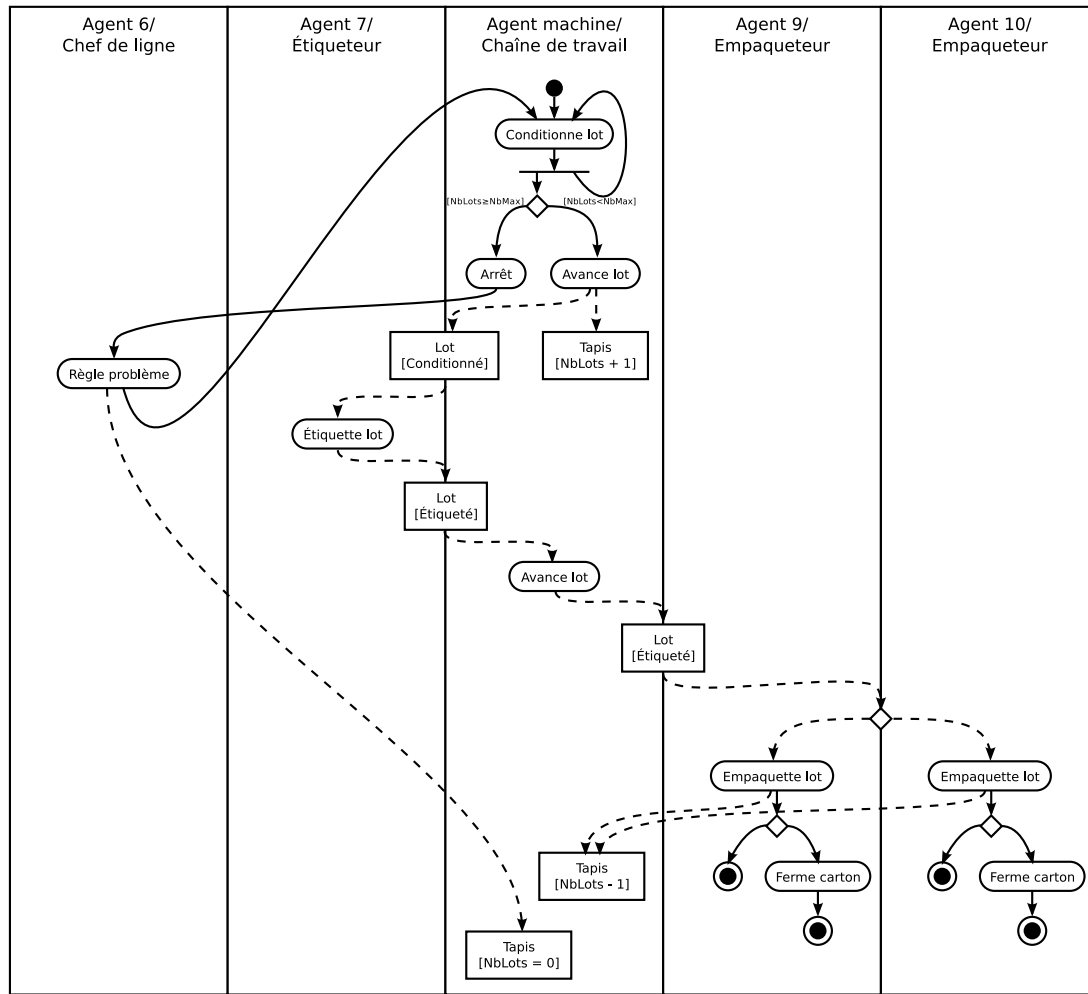


FIG. 4.15 – Diagramme d’activités du fonctionnement normal de la chaîne.

port à une tâche en commun (poste 9/10)). Ce diagramme est complété par des diagrammes plus précis correspondant à chaque poste et permettant de faire figurer des comportements plus particuliers. La figure 4.16 donne l’exemple du poste 7.

Ce diagramme représente une vue détaillée du diagramme de la figure 4.15. Il ne comporte que la machine, l’avatar 7 et le chef de ligne qui intervient dans la résolution des problèmes. On voit que l’agent 7 peut avoir à faire à des lots plus ou moins bien conditionnés⁷⁵. Devant un lot bien conditionné, il peut choisir d’accomplir sa tâche normale (étiqueter le lot) ou bien de ne rien faire. Devant un lot mal conditionné, il peut l’étiqueter quand même, enlever le lot, jeter le lot et dans tous les cas choisir ou non de prévenir le chef de ligne. Chacun de ces choix⁷⁶

⁷⁵La qualité de cette évaluation peut elle-même dépendre de l’état émotionnel et psychologique de l’individu.

⁷⁶Choix ou plutôt comportements, l’action accomplie n’étant pas entièrement délibérée puisqu’elle dépend en partie de composantes de l’état interne (émotionnelles, psychologiques, ...).

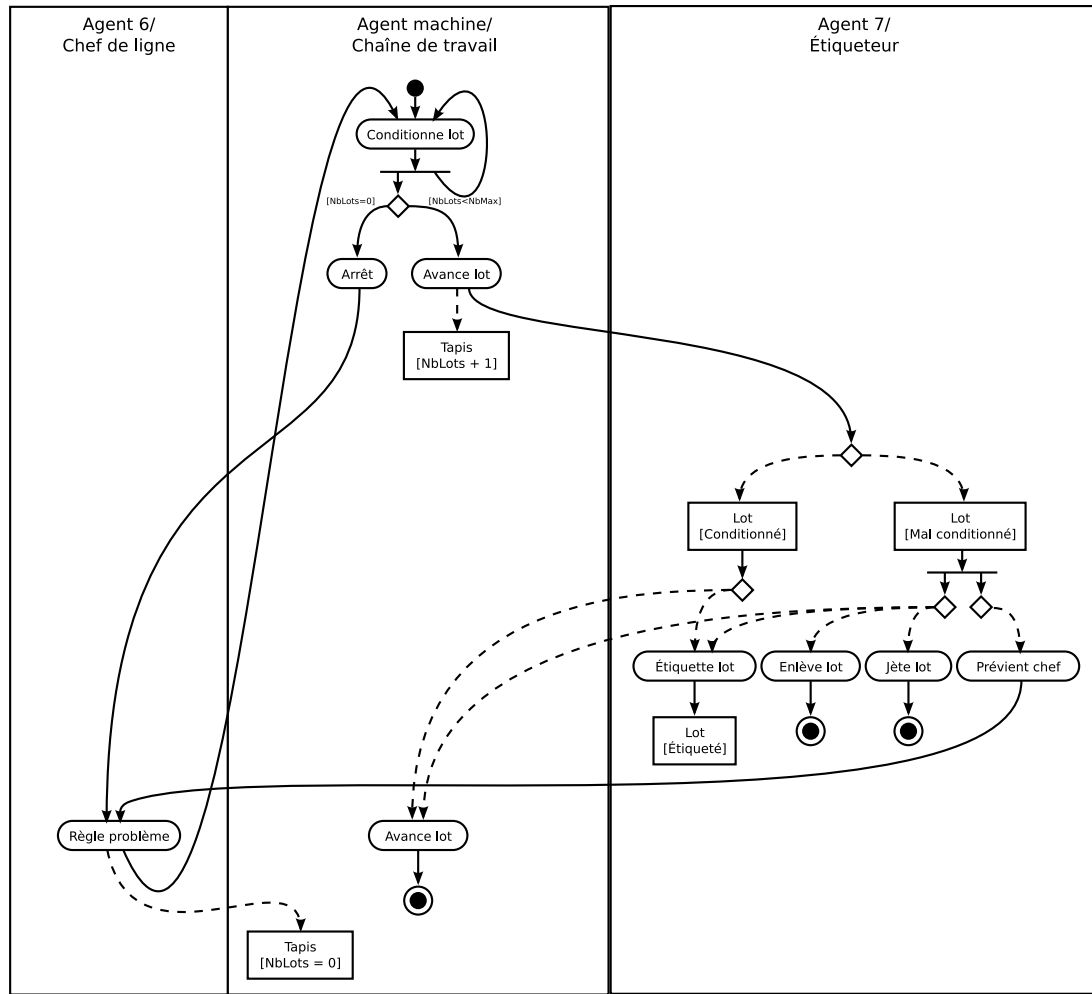


FIG. 4.16 – Diagramme d'activités du poste 7.

est conditionné par des règles comportementales. Une version préliminaire de ces règles a été établie par des psychologues. Le diagramme de séquence fournit alors une vue sous un autre angle du déroulement d'une tâche permettant de faire figurer plus de détails sur les conditions de déclenchement de chaque action.

Sur ce diagramme, nous avons utilisé les notations proposées par AUML pour fournir une meilleure lisibilité. Ce diagramme correspond à l'exécution d'une tâche de travail sur le poste 7 dans le cas où l'agent n'a qu'un paquet dans sa zone de travail (situation normale). C'est en fait une partie de l'instance du protocole *travail* présenté dans la partie 4.2.2, le protocole complet présenté en annexe prenant également en compte les cas où il n'y a pas de paquet et où il y a trop de paquets. Nous avons donc utilisé ici plusieurs structures conditionnelles imbriquées. Au niveau le plus haut, on distingue les cas *bien scellé* et *mal scellé*. Ensuite, suivant ces deux cas, on retrouve les tâches du diagramme 4.16, l'avantage est qu'ici on peut spécifier des conditions

à l'exécution de chacune de ces tâches (si la personne est négligente et détendue, elle laissera le paquet sans rien faire). On voit également que la notation *ref* permet d'imbriquer un autre diagramme de séquence, ici le diagramme des appels (appel du chef de ligne), à celui-ci.

Communication entre agents

Nous allons présenter dans cette partie le détail de l'interaction de type demande d'aide entre agents. Comme dans la partie précédente, nous avons d'abord établi un diagramme d'activités complété par un diagramme de séquence permettant de fournir plus de détails. Ce diagramme de séquence est en fait une instance du protocole *demande d'aide* présenté en partie 4.2.2 où nous avons précisé les conditions de chacun des envois de messages entre agents. Ces diagrammes possèdent les mêmes caractéristiques que ceux que nous avons présentés dans la partie précédente, aussi ne détaillerons nous pas leur fonctionnement. Il est cependant important de préciser que, dans le cas de la communication entre agents, les caractéristiques conditionnant le comportement sont légèrement différentes de celles dans le travail. Les interactions dépendent donc ici d'un ensemble de règles impliquant entre autres les sentiments et émotions relatives que peut avoir un individu envers un autre (du type amitié, rancœur, haine). Des psychologues et le travail d'expertise sur les interactions comportementales déjà effectué pour PROJEQUIP sont utilisés pour établir ces règles.

Conclusion

Nous avons présenté dans ce chapitre notre modèle d'agent émotionnel EFT en le déclinant à la fois sur le modèle internet de l'agent et sur le modèle des interactions entre entités du système. Ce modèle, basé sur une architecture *BDI*, se distingue par sa structure en 3 couches au niveau de l'affect où nous distinguons les émotions, les sentiments et le tempérament. Chacune de ces facettes de l'affect se distingue par la rémanence et l'intensité des variations dans le temps. Cette complémentarité nous permet, à l'inverse d'autres modèles se focalisant sur les émotions, de doter notre agent d'un comportement individuel unique et convaincant. La prise en compte de ces dimensions affectives à de nombreux niveaux dans les processus décisionnels de l'agent grâce au système de base de règles que nous avons mis en place est également essentiel afin de rendre un comportement cohérent. Nous avons d'autre part détaillé la modélisation des interactions entre agents encore une fois en prenant en compte les émotions et leur expression dans la communication entre entités. Cette modélisation a été déclinée dans le cadre applicatif qui nous intéresse. L'ensemble de ces modèles ont été réalisés grâce à AgentUML, l'initiative de la FIPA afin de proposer une évolution de la modélisation UML adaptée aux agents et aujourd'hui intégrée à la version 2.0 de ce standard.

5

Un modèle de fouille de trace d'agents

Ce chapitre décrit le processus de fouille de données que nous avons mis en œuvre pour l'étude des données produites par la simulation. Y sont décrits d'une part les méthodes de pré-traitement des données et d'autre part les méthodes de fouilles proprement dites.

Sommaire

5.1	Données issues de la simulation	128
5.1.1	Des données propres	128
5.1.2	Des données malléables	128
5.1.3	Des données à volonté	129
5.2	Pré-traitement des données	129
5.2.1	Discrétisation des données	129
5.2.2	Dissimilarité entre vecteurs	130
5.2.3	Réduction dimensionnelle	137
5.2.4	Agrégation de vecteurs	137
5.3	Méthode des projections aléatoires	141
5.3.1	Présentation générale de la méthode	141
5.3.2	Étapes du traitement	141
5.3.3	Mise en oeuvre de la méthode	143

Introduction

Nous avons vu dans les chapitres précédents toute la spécificité des données issues de simulation : leur volumétrie, leur aspect multidimensionnel à la fois dans présent dans les données issues d'une unique itération de la simulation et dans celles issues de plusieurs itération qu'il serait nécessaire de croiser. Notre objectif étant de déceler les séquences d'évènements caractéristiques survenant au cours d'une simulation, nous sommes appuyés sur les travaux que nous avons cité dans les parties précédentes pour élaborer une méthode de fouille adaptée à nos données. Nous allons ainsi détailler cette méthode appliquées aux traces d'un agent permettant de

détecter qu'un état A à un instant donnée sera généralement suivi des états B , C et D quelques instants plus tard.

Nous présenterons donc dans un premier temps l'intérêt des données de simulation cette fois-ci selon le point de vue de la fouille et notamment du pré-traitement des données largement facilité pour ce type de données. Nous présenterons ensuite le pré-traitement que nous avons mis en place pour permettre à notre algorithme de traiter les données. Nous présenterons enfin notre méthode de fouille librement inspirée de la méthode des projections aléatoires.

5.1 Données issues de la simulation

Comme nous l'avons déjà présenté dans la partie 3.2, la simulation et les données qu'elle peut générer possèdent plusieurs particularités par rapport aux données communément étudiées dans en ECD.

5.1.1 Des données propres

La première particularité est la qualité des données. En effet, contrairement au recueil effectué en milieu réel souvent sujet à des altérations pouvant entraîner des données bruitées ou manquantes, les données issues d'une simulation sont « propres ». Si ce n'est pas le cas parce que la collecte s'est mal déroulée, il suffit de relancer la simulation pour avoir un nouveau jeu d'essai de qualité. Cela dispense donc complètement dans l'étape de prétraitement des phases de pré-traitement et de gestion des valeurs manquantes classiquement menées en ECD.

5.1.2 Des données malléables

D'autre part, le fait que l'expérimentateur ai la main sur l'origine des données rend possible un confort inaccessible dans l'étude de phénomène réel. On peut ainsi choisir la fréquence d'enregistrement des données, l'ajout ou la suppression de la journalisation de certains paramètres, le format d'enregistrement des données de manière très flexible. Chaque nouvelle itération de la simulation peut être paramétrée selon les désirs de l'expérimentateur. Ainsi, chose originale par rapport au processus de fouille classique, les phases de supervision et de raffinement respectivement menées pendant la recherche de règles et après la génération des règles candidates pour n'en sélectionner que les plus pertinentes peuvent être menées également durant la génération des données à étudier. Cela offre un luxe supplémentaire et la possibilité de parvenir à un paramétrage très fin pouvant permettre d'étudier et d'extraire exactement les connaissances désirées.

5.1.3 Des données à volonté

La simulation est un processus totalement sous contrôle de l'expérimentateur et, c'est là tout son intérêt en informatique, quasiment gratuite hormis son coût en calcul et en temps d'exécution. Libre donc à lui de l'exécuter autant de fois qu'il le désire et de collecter ainsi autant de données qu'il le désire. Les cas d'études peuvent donc être aussi nombreux qu'il est nécessaire et la représentativité statistique des données assurée. Néanmoins, cette liberté amène une nouvelle problématique : comment tirer efficacement parti de ces multiples données issues d'itérations différentes d'une simulation en effectuant une fouille non pas interne à chaque échantillon récoltés mais transverse à l'ensemble des échantillons. On entrevoit ainsi la possibilité de fournir à l'expérimentateur l'ensemble des scénarii possible à partir d'un cas initial déterminé comme nous en avons parlé dans la partie 3.2 et d'en extraire ceux bénéficiant du meilleur support.

5.2 Pré-traitement des données

Nous allons présenter dans cette partie un certain nombre de méthodes de pré-traitement des données qu'il nous a été nécessaire de mettre en œuvre pour ensuite réaliser la fouille de données proprement dite. Le but de cette étape de pré-traitement est de réaliser une abstraction des données.

L'abstraction est une représentation de « haut niveau » qui, plus qu'une simple transformation de l'espace ou simple réduction de dimension, inclut l'interprétation des données pour leur donner un sens et un niveau de détail adaptés à l'objectif de leur analyse. Duchêne (2004)

. L'objectif est donc de transformer le niveau de granularité des données pour l'étude. Quand il s'agit de données temporelles, il s'agit principalement de gommer les petites variations pour ne conserver que les tendances principales. Dans le domaine du traitement du signal, on parlerait alors de l'application d'un filtre passe-bas aux données. Dans notre cas, les données étant relativement contrôlées puisqu'issues d'une simulation, nous n'avons pas à traiter les phénomènes de bruit présent dans la plupart des données réelles (issues de capteurs notamment). Nous avons donc appliqué deux traitements sur les données :

- **Discrétisation** des données quantitatives pour obtenir une représentation homogène des données initialement hétérogènes,
- **Agrégation** des séquences de vecteurs ramenant une séquence de vecteurs différents à un seul vecteur quand ceux-ci sont suffisamment similaires.

5.2.1 Discrétisation des données

La première étape du pré-traitement consiste à discrétiser les valeurs numériques. Le sous-échantillonnage est une méthode qui permet d'atteindre ce but : appliquée aux valeurs d'un attribut, elle vise à en réduire le nombre. On rencontre alors deux problématiques : la méthode

de définition des intervalles de discrétisation et la méthode de sous-échantillonnage employées.

La définition des intervalles de discrétisation peut être déterminée de manière empirique par l'utilisateur, ou au contraire reposer sur des contraintes spécifiques liées notamment à la distribution des valeurs d'un attribut (équiprobabilité des symboles (Lin et al., 2002a; Chiu and Keogh, 2003), variance maximum pour les valeurs associées à chaque symbole (Hong and Huang, 2000) ou encore sur une classification supervisée ou non des valeurs possibles (Das et al., 1997; Hong and Huang, 2000)). Dans notre cas, et contrairement à Lin et al. (2002a), les distributions de valeurs n'ont aucune raison de suivre une loi quelconque. Aussi, avons nous choisi une méthode simple de sous-échantillonnage où l'utilisateur précise au départ le nombre n maximum de valeurs qu'il souhaite avoir pour un attribut. La plage de variabilité d'un attribut entre son minimum et son maximum peut ensuite être découpée en $n - 1$ intervalles dont les bornes sont les valeurs cibles du sous-échantillonnage. On parcourt ensuite l'ensemble des valeurs de l'attribut, on détermine auquel des intervalles précédemment calculés elle appartient et suivant sa position dans l'intervalle, on lui substitue la valeur supérieure ou inférieure de l'intervalle. La figure 5.1 schématise le processus détaillé dans l'algorithme 5.1.

Ce sous-échantillonnage n'est pas influencé par la distribution des valeurs d'un attribut. En effet, nous aurions également pu utiliser le sous-échantillonnage par quantile ou une méthode de classification. L'avantage de ce type de méthodes est qu'elles privilégient les zones de valeurs denses (*i.e.* fortement rapprochées) en y associant plus de classes que sur les zones moins denses. Imaginons par exemple qu'un attribut est une distribution gaussienne, l'échantillonnage sera plus fin pour les valeurs centrales que pour les extrêmes. Néanmoins, cela peut également apporter un biais en entraînant une sur-représentation des variations sur les zones où les valeurs sont denses (et réciproquement sur les zones peu denses) non souhaitable. Nous avons donc pris le parti de traiter tout attribut avec un échantillonnage uniforme sur sa plage de valeurs.

5.2.2 Dissimilarité entre vecteurs

Mesure de dissimilarité

Nous allons devoir disposer pour les étapes suivantes (réduction dimensionnelle et agrégation dans une moindre mesure) d'une mesure de similarité entre vecteurs (ou plutôt dans notre cas de dissimilarité). Ces mesures sont nombreuses dans la littérature et le choix est difficile puisque fortement lié aux données cibles. La particularité de nos données est qu'elles sont hétérogènes, à la fois quantitatives et qualitatives que nous ne pouvons pas ramener à des données quantitatives car elles ne sont pas ordonnables. L'ensemble des vecteurs ne définit donc pas un espace vectoriel (pas d'additivité et produit scalaire définissables sur les données qualitatives). La distance euclidienne classique entre deux vecteurs n'est donc pas applicable. La distance de Hamming (1950), largement utilisée en informatique, nous avait paru dans un premier temps un choix judicieux. En effet, cette distance calcule le cardinal de l'ensemble des différences entre deux ensembles de valeurs. Dans notre cas, cela nous donnera donc si nous prenons les deux vecteurs exemples suivants : [*Joie* = 3, *Extraversion* = 8, *Bouche* = *sourire*] et [*Joie* = 0, *Extraversion* = 8, *Bouche* = *neutre*], une distance de 2. Si cette distance est effec-

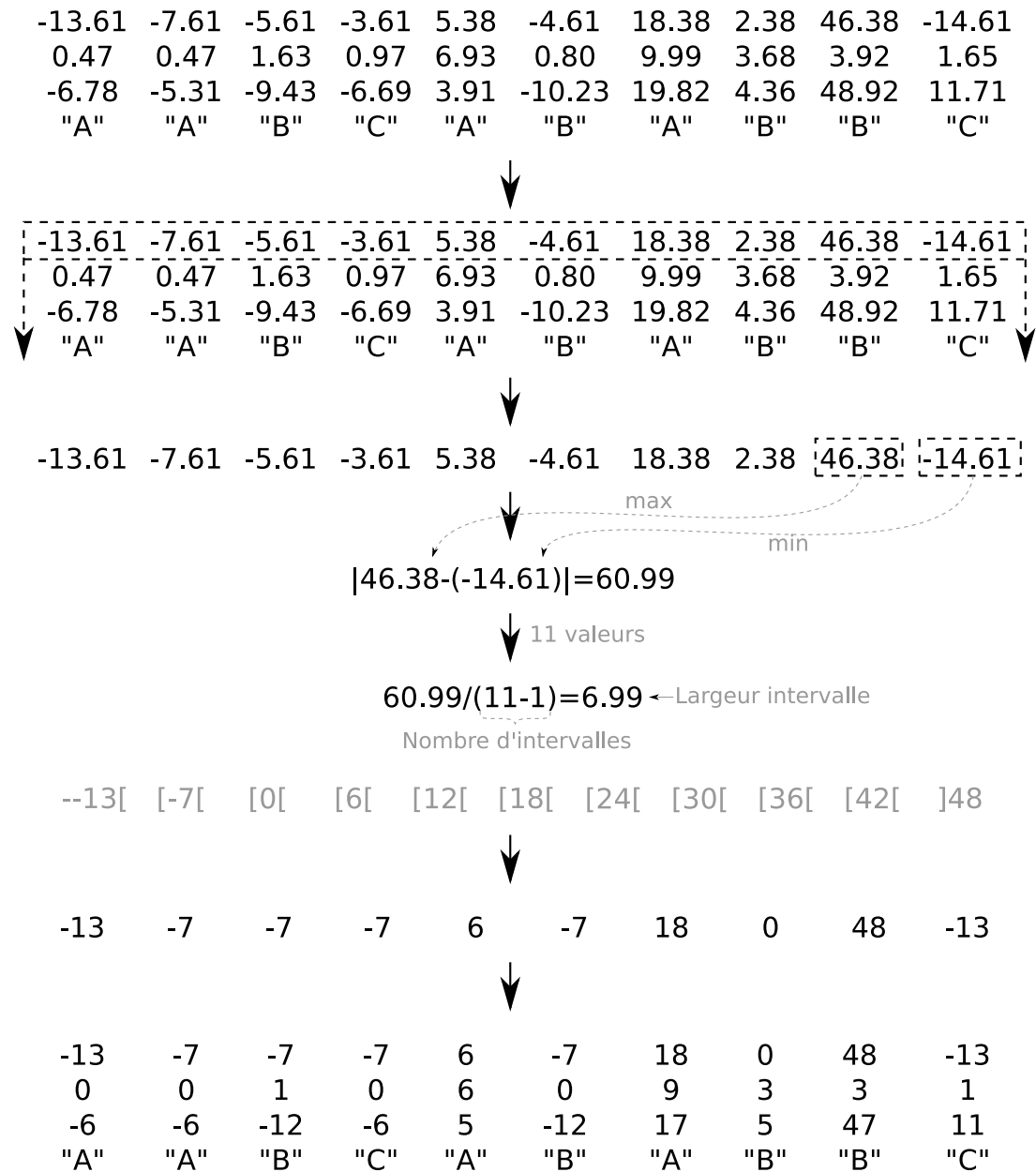


FIG. 5.1 – Illustration de la procédure de sous-échantillonnage.

Algorithme 5.1 Algorithme de sous-échantillonnage.

```

1: Fonction SOUS-ECHANTILLONAGE( $\mathbb{X}$ ,  $v$ )
2:    $\triangleright \mathbb{X}$  : une série temporelle de vecteurs  $X_i = \{x_{1_{X_i}}, x_{2_{X_i}}, \dots, x_{n_{X_i}}, q_{1_{X_i}}, q_{2_{X_i}}, \dots, q_{m_{X_i}}\}$ 
3:    $\triangleright v$  : le nombre de valeurs de sous-échantillonnage souhaité
4:    $\triangleright$  Retourne  $\mathbb{X}'$  : la série temporelle sous-échantillonnée

  Initialisation
5:    $nb_{vects} \leftarrow$  taille de  $\mathbb{X}$ 
6:    $\mathbb{X}' \leftarrow$  nouveau vecteur de taille  $nb_{vects}$ 
7:    $nb_{attrs} \leftarrow$  nombre d'attributs d'un vecteur  $X_i$  de  $\mathbb{X}$   $\triangleright n + m$  ici

  Traitement
8:   Pour  $i$  allant de 1 à  $nb_{attrs}$  faire
9:     Si l'attribut  $x_{i_{X_k}}$   $k \in \{1, nb_{vects}\}$  est quantitatif alors
10:        $valeurs_{attr} \leftarrow x_{i_{X_k}}$   $k \in \{1, nb_{vects}\}$ 
11:        $min_{attr} \leftarrow \min(valeurs_{attr})$ 
12:        $max_{attr} \leftarrow \max(valeurs_{attr})$ 
13:        $longueur_{intervalles} \leftarrow E \left( \frac{|max_{attr} - min_{attr}|}{v-1} \right)$ 
14:       Pour  $j$  allant de 1 à  $nb_{vects}$  faire  $\triangleright$  Pour toutes les valeurs de l'attribut sur
          l'ensemble des vecteurs de  $\mathbb{X}$ 
15:          $\triangleright$  Diviser la valeur par la longueur des intervalles
16:          $quotient \leftarrow \frac{x_{i_{X_j}}}{longueur_{intervalles}}$ 
17:         Si  $quotient - E(quotient) < 0,5$  alors  $\triangleright$  Si partie décimale  $< 0.5$ 
18:            $\triangleright$  Dans  $\mathbb{X}'$ , on met la borne inférieure de l'intervalle
19:            $x_{i_{X'_j}} \leftarrow E(quotient) * longueur_{intervalles}$ 
20:         Sinon
21:            $\triangleright$  Dans  $\mathbb{X}'$ , on met la borne supérieure de l'intervalle
22:            $x_{i_{X'_j}} \leftarrow (E(quotient) + 1) * longueur_{intervalles}$ 
23:         Fin Si
24:       Fin Pour
25:     Sinon
26:        $\triangleright$  Dans  $\mathbb{X}'$ , on met la valeur de l'attribut quantitatif telle quelle
27:        $x_{i_{X'_j}} \leftarrow x_{i_{X_j}}$ 
28:     Fin Si
29:   Fin Pour
30:   Retourner  $\mathbb{X}'$   $\triangleright$  La série temporelle sous-échantillonnée
31: Fin Fonction

```

tivement intéressante, on s'aperçoit tout de suite qu'elle perd une partie des informations disponibles lorsqu'on l'applique à des données quantitatives, puisqu'elle vaut systématiquement 1 si les deux valeurs sont différentes quelle que soit leur distance euclidienne. Afin de travailler avec une mesure plus pertinente, nous avons donc décidé de créer une mesure de dissimilarité qui soit un compromis entre la distance euclidienne et la distance de Hamming.

Nous avons ainsi défini une mesure basée sur une distance euclidienne normée pour les valeurs quantitatives et sur une distance de Hamming pour les valeurs qualitatives. Soit \mathbb{X} la série temporelles de vecteurs telle que $X = \{x_{1X}, x_{2X}, \dots, x_{nX}, q_{1X}, q_{2X}, \dots, q_{mX}\} \in \mathbb{X}$ est un vecteur tel que $x_{iX} \in \mathbb{R}$ et q_{iX} soit une donnée quantitative. Pour chacun des attributs quantitatifs du vecteur, on le norme pour qu'il soit compris entre 0 et 1 :

$$\forall X \in \mathbb{X} | X = \{x_{1X}, x_{2X}, \dots, x_{nX}, q_{1X}, q_{2X}, \dots, q_{mX}\},$$

$$\|X\| = \left\{ \frac{x_{1X} - \min_{\mathbb{X}}(x_1)}{\max_{\mathbb{X}}(x_1) - \min_{\mathbb{X}}(x_1)}, \frac{x_{2X} - \min_{\mathbb{X}}(x_2)}{\max_{\mathbb{X}}(x_2) - \min_{\mathbb{X}}(x_2)}, \dots, \frac{x_{nX} - \min_{\mathbb{X}}(x_n)}{\max_{\mathbb{X}}(x_n) - \min_{\mathbb{X}}(x_n)}, q_{1X}, q_{2X}, \dots, q_{mX} \right\}.$$

Le traitement permettant de normer les vecteurs de la série est présenté dans l'algorithme 5.2.

On définit alors la distance euclidienne \mathcal{D}_{qual} entre valeurs quantitatives par :

$$\forall x_1, x_2 \in \mathbb{R}^2, \mathcal{D}_{qual} = |x_2 - x_1|$$

et la distance de Hamming \mathcal{D}_{quant} entre deux valeurs qualitatives. Nous définissons alors la mesure de dissimilarité \mathcal{S} entre deux vecteurs de \mathbb{X} par :

$$\forall X, Y \in \mathbb{X}^2 | X = \{x_{1X}, x_{2X}, \dots, x_{nX}, q_{1X}, q_{2X}, \dots, q_{mX}\} \&$$

$$Y = \{x_{1Y}, x_{2Y}, \dots, x_{nY}, q_{1Y}, q_{2Y}, \dots, q_{mY}\},$$

$$\mathcal{S}(X, Y) = \frac{\sum_{i=1}^n \mathcal{D}_{quant}(x_{iX}, x_{iY})}{n} + \frac{\sum_{i=1}^m \mathcal{D}_{qual}(q_{iX}, q_{iY})}{m}.$$

La mise en oeuvre de cette mesure est présentée dans l'algorithme 5.3.

Le cas particulier des données spatiales

Les données spatiales sont un type de données dont les spécificités ajoutent à la complexité du calcul de distance. En effet, si on se place dans un repère bidimensionnel pour prendre le cas le plus simple, la position de tout objet assimilable à un point est représentée par le couple de coordonnées de son centre de gravité (x, y) . Les mesures de distance entre deux objets sont communes, mais la difficulté réside plus dans la façon de stocker cette donnée. En effet, alors que la plupart des autres données sont stockées sous la forme *attribut = valeur*, il serait nécessaire de stocker celle-ci sous la forme *attribut = (coordonnee_x, coordonnee_y)* et

Algorithme 5.2 Algorithme de calcul de la norme.

```

1: Fonction NORME( $\mathbb{X}$ )
2:   ▷  $\mathbb{X}$  : une série temporelle de vecteurs  $X_i = \{x_{1X_i}, x_{2X_i}, \dots, x_{nX_i}, q_{1X_i}, q_{2X_i}, \dots, q_{mX_i}\}$ 
3:   ▷ Retourne  $\mathbb{X}'$  : la série temporelle normée
4:    $nb_{vects} \leftarrow$  taille de  $\mathbb{X}$ 
5:    $\mathbb{X}'' \leftarrow$  nouveau vecteur de taille  $nb_{vects}$ 
6:    $nb_{attrs} \leftarrow$  nombre d'attributs d'un vecteur  $X_i$  de  $\mathbb{X}$  ▷  $n + m$  ici
7:    $vecteur_{maxs} \leftarrow$  nouveau vecteur de taille  $nb_{attrs}$ 
8:    $vecteur_{mins} \leftarrow$  nouveau vecteur de taille  $nb_{attrs}$ 
9:   Pour  $i$  allant de 1 à  $nb_{attrs}$  faire
10:    Si l'attribut  $x_{iX_k}$   $k \in \{1, nb_{vects}\}$  est quantitatif alors
11:       $valeurs_{attr} \leftarrow x_{iX_k}$   $k \in \{1, nb_{vects}\}$ 
12:       $vecteur_{maxs}[i] \leftarrow \max(valeurs_{attr})$ 
13:       $vecteur_{mins}[i] \leftarrow \min(valeurs_{attr})$ 
14:    Fin Si
15:  Fin Pour
16:  Pour  $j$  allant de 1 à  $nb_{vects}$  faire
17:    Pour  $i$  allant de 1 à  $nb_{vects}$  faire
18:      Si l'attribut  $x_{iX_j}$  est quantitatif alors
19:        Si  $vecteur_{maxs}[i] \neq vecteur_{mins}[i]$  alors
20:           $x_{iX'_j} \leftarrow \frac{x_{iX_j} - vecteur_{mins}[i]}{|vecteur_{mins}[i] - vecteur_{maxs}[i]|}$ 
21:        Sinon
22:           $x_{iX'_j} \leftarrow 0$ 
23:        Fin Si
24:      Sinon
25:         $x_{iX'_j} \leftarrow x_{iX_j}$ 
26:      Fin Si
27:    Fin Pour
28:  Fin Pour
29:  Retourner  $\mathbb{X}''$ 
30: Fin Fonction

```

Algorithme 5.3 Algorithme de mesure de la dissimilarité.

```

1: Fonction MESURE-DISSIMILARITE-ATTRIBUTS( $x_1, x_2$ )
2:   ▷  $x_1, x_2$  : deux attributs de deux vecteurs normés
3:   ▷ Retourne la distance entre les 2 attributs
4:   Si  $x_1$  est quantitatif alors
5:     Retourner  $|x_1 - x_2|$ 
6:   Sinon
7:     Si  $x_1 = x_2$  alors
8:       Retourner 0
9:     Sinon
10:      Retourner 1
11:   Fin Si
12: Fin Si
13: Fin Fonction

14: Fonction MESURE-DISSIMILARITE( $X_1, X_2$ )
15:   ▷  $X_1, X_2$  : deux vecteurs (au préalable normés)
16:   ▷ Retourne la distance entre les 2 vecteurs
17:    $nb_{attrs} \leftarrow$  nombre d'attributs d'un vecteur  $X_i$ 
18:   Retourner  $\frac{\sum_{i=1}^{nb_{attrs}} \text{MESURE-DISSIMILARITE-ATTRIBUTS}(x_{iX_1}, x_{iX_2})}{nb_{attrs}}$ 
19: Fin Fonction

```

traiter spécifiquement cet attribut pour le calcul de distance. Une autre approche envisageable serait d'effectuer un pré-traitement pour traduire l'information de position en elle-même peu informative en une ou plutôt plusieurs informations de proximité entre les entités du système à un instant donné. En effet, il paraît assez évident que ce n'est pas la position absolue de l'agent dans un référentiel mais plutôt sa position relative vis-à-vis des autres entités du système qui peut avoir un impact sur son comportement. Bien que ça ne soit pas impossible, nous n'avons pas traité ce point dans nos travaux.

Matrice de dissimilarité

Pour compléter le calcul de dissimilarité entre deux vecteurs de la suite temporelle et effectuer les traitements suivants (notamment la réduction dimensionnelle), il est nécessaire de calculer l'ensemble des distances entre les vecteurs composant la suite. Nous avons pour cela utilisé une classique matrice de dissimilarité obtenu grâce à l'algorithme 5.4. Le résultat de ce calcul sur les données déjà présentées en figure 5.1 est présenté dans la figure 5.2.

Algorithme 5.4 Algorithme de calcul de la matrice de dissimilarité.

```

1: Fonction MATRICE-DISSIMILARITE( $\mathbb{X}$ )
2:   ▷  $\mathbb{X}$  : une série temporelle de vecteurs  $X_i = \{x_{1_{X_i}}, x_{2_{X_i}}, \dots, x_{n_{X_i}}, q_{1_{X_i}}, q_{2_{X_i}}, \dots, q_{m_{X_i}}\}$ 
3:   ▷ Retourne la matrice de dissimilarité entre l'ensemble des vecteurs de  $\mathbb{X}$ 
4:    $\mathbb{X}' \leftarrow \text{NORME}(\mathbb{X})$ 
5:    $nb_{vects} \leftarrow \text{taille de } \mathbb{X}$ 
6:    $matrice \leftarrow$  nouvelle matrice de dimension  $nb_{vects} * nb_{vects}$  remplie de 0
7:   ▷ Remplissage de la matrice, on ne traite que la partie triangulaire haute puisque la
   matrice est symétrique et que sa diagonale est remplie de 0
8:   Pour  $i$  allant de 1 à  $nb_{vects} - 1$  faire
9:     Pour  $j$  allant de 2 à  $nb_{vects}$  faire
10:       $matrice[i, j] \leftarrow \text{MESURE-DISSIMILARITE}(X_i, X_j)$ 
11:     Fin Pour
12:   Fin Pour
13:   Retourner  $matrice$ 
14: Fin Fonction

```

	V1	V2	V3	V4	V5	V6	V7	V8	V9	V10
V1	0.00	0.03	0.32	0.30	0.29	0.31	0.49	0.45	0.82	0.36
V2	0.03	0.00	0.31	0.29	0.26	0.29	0.46	0.42	0.79	0.38
V3	0.32	0.31	0.00	0.29	0.49	0.03	0.69	0.14	0.52	0.38
V4	0.30	0.29	0.29	0.00	0.49	0.27	0.69	0.39	0.77	0.14
V5	0.29	0.26	0.49	0.49	0.00	0.51	0.20	0.35	0.69	0.50
V6	0.31	0.29	0.03	0.27	0.51	0.00	0.71	0.17	0.54	0.41
V7	0.49	0.46	0.69	0.69	0.20	0.71	0.00	0.55	0.65	0.64
V8	0.45	0.42	0.14	0.39	0.35	0.17	0.55	0.00	0.37	0.40
V9	0.82	0.79	0.52	0.77	0.69	0.54	0.65	0.37	0.00	0.72
V10	0.36	0.38	0.38	0.14	0.50	0.41	0.64	0.40	0.72	0.00

FIG. 5.2 – Matrice de dissimilarité sur les données de la figure 5.1.

5.2.3 Réduction dimensionnelle

L'ordonnancement des deux étapes suivante peut-être inversé. En effet, nous procédons à une réduction des données à une dimension ainsi qu'à une agrégation des vecteurs consécutifs identiques. La réduction dimensionnelle de la série temporelle consiste à associer un symbole à chaque vecteur différent afin d'appliquer des algorithmes de traitement de données mono-dimensionnelles à notre jeux de données. Pour distinguer les vecteurs différents, on doit donc utiliser une mesure de similarité. L'agrégation consiste à regrouper en un seul vecteur une séquence de vecteurs identiques (nous verrons dans la section suivante l'intérêt de cette opération). Là encore, il est nécessaire d'utiliser une mesure de similarité. Effectuer la réduction avant l'agrégation permet de diminuer légèrement le temps de traitement puisque la mesure de similarité utilisée pour l'agrégation se résume à une comparaison de symboles (de manière sous-jacente, on utilise en fait la même mesure de similarité que pour la réduction). Dans l'ordre inverse, le temps de traitement est plus long mais cela peut permettre éventuellement d'utiliser une mesure de similarité différente (notamment au niveau des seuils de similarité) pour chaque opération. Nous avons préféré privilégié le temps de traitement, des mesures de similarité différente n'ayant pas forcément d'intérêt dans notre cas.

Pour procéder à la réduction à une dimension, nous nous appuyons sur la matrice de dissimilarité. Après avoir fixé un seuil au dessous duquel deux vecteurs sont considérés comme identiques, nous parcourons la matrice pour associer à les symboles aux vecteurs. Le processus est itératif. L'utilisateur fixe un seuil de similarité. On trouve alors l'ensemble des vecteurs similaires au premier de la série et on leur associe un symbole identique, un nouveau symbole est associé à tout vecteur rencontré différent du premier. On procède de la même manière pour le deuxième vecteur, en remplaçant les symboles précédemment associés aux vecteurs rencontrés si ceux-ci sont identiques au second. On itère ce traitement jusqu'à arriver au dernier vecteur de la série. La figure 5.3 illustre ce principe. Sa mise en oeuvre est détaillée dans l'algorithme 5.5.

5.2.4 Agrégation de vecteurs

La dernière phase du pré-traitement des données consiste à agréger les vecteurs « similaires » se suivant dans le temps. Cette étape n'est pas indispensable, on pourrait en effet conserver les répétitions de phénomènes similaires se succédant dans le temps puisque ceux-ci peuvent avoir une signification intéressante. Le problème rencontrer alors est que ces suite de vecteurs similaires vont fortement polluer la recherche de motifs fréquents : on risque de ne trouver que ces suites de vecteurs similaires en motifs fréquents si elles sont trop nombreuses. C'est pourquoi nous avons préféré dans un premier temps procéder à une agrégation des vecteurs. Il serait sans doute judicieux dans un second temps d'élargir la méthode pour qu'elle puisse gérer sans les problèmes de pollution évoqués les données répétitives. Avec le choix d'ordonnancement des phases que nous avons fait, la procédure d'agrégation devient extrêmement simple comme l'illustre l'algorithme 5.6.

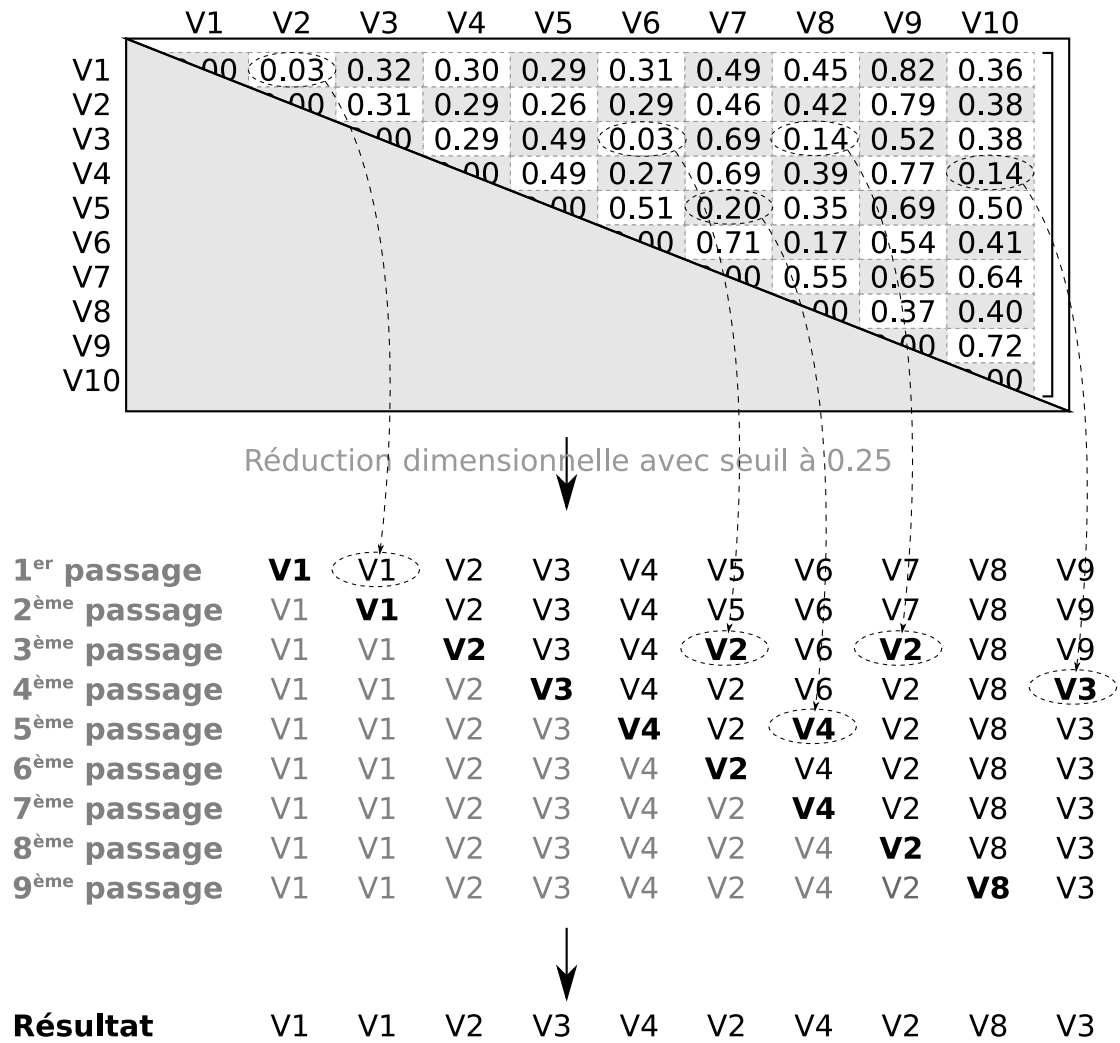


FIG. 5.3 – Illustration de la procédure de réduction dimensionnelle.

Algorithme 5.5 Algorithme de réduction des données.

```
1: Fonction REDUIRE( $\mathbb{X}$ , seuil)
2:   ▷  $\mathbb{X}$  : une série temporelle de vecteurs  $X_i = \{x_{1_{X_i}}, x_{2_{X_i}}, \dots, x_{n_{X_i}}, q_{1_{X_i}}, q_{2_{X_i}}, \dots, q_{m_{X_i}}\}$ 
3:   ▷ seuil : le seuil au dessous duquel 2 vecteurs sont considérés comme similaires
4:   ▷ Retourne  $V$  : un vecteur de symboles, série temporelle réduite à 1 dimension
5:    $nb_{vects} \leftarrow$  taille de  $\mathbb{X}$ 
6:    $V \leftarrow$  nouveau vecteur de dimension  $nb_{vects}$ 
7:    $matrice \leftarrow$  MATRICE-DISSIMILARITE( $\mathbb{X}$ )
8:   Pour  $i$  allant de 1 à nombre de lignes de matrice faire
9:     Pour  $j$  allant de 1 à nombre de colonnes de matrice  $-1$  faire
10:      Si  $i = j$  alors
11:         $V[j] \leftarrow$  nouveau symbole
12:      Sinon Si  $i \neq j$  et  $matrice[i, j] < \textit{seuil}$  alors
13:         $V[j] \leftarrow V[i]$ 
14:      Sinon
15:        Si  $i = 1$  alors
16:           $V[j] \leftarrow$  nouveau symbole
17:        Fin Si
18:      Fin Si
19:    Fin Pour
20:  Fin Pour
21:  Retourner  $V$ 
22: Fin Fonction
```

Algorithme 5.6 Algorithme d'agrégation des données réduites.

```

1: Fonction AGREGER( $V$ )
2:   ▷  $V$  : un vecteur de symboles, série temporelle réduite à 1 dimension
3:   ▷ Retourne  $V'$  : un vecteur de symboles agrégé
4:    $V' \Leftarrow V$ 
5:    $decalage \Leftarrow 0$ 
6:   Pour  $i$  allant de 1 à taille de  $V - 1$  faire
7:     Si  $V[i] = V[i + 1]$  alors
8:        $V_{gauche} \Leftarrow \text{SOUS-ENSEMBLE}(V', 1, i - decalage)$ 
9:        $V_{droite} \Leftarrow \text{SOUS-ENSEMBLE}(V', i - decalage + 2, \text{taille de } V')$ 
10:       $V' \Leftarrow \text{CONCATENER}(V_{gauche}, V_{droite})$ 
11:     Fin Si
12:   Fin Pour
13:   Retourner  $V'$ 
14: Fin Fonction

15: Fonction CONCATENER( $V1, V2$ )
16:   ▷ Fonction permettant de concaténer deux vecteurs
17: Fin Fonction

18: Fonction SOUS-ENSEMBLE( $V, i, j$ )
19:   ▷ Fonction permettant d'extraire le sous-vecteur de  $V$  entre les bornes  $i$  et  $j$ 
20: Fin Fonction

```

5.3 Méthode des projections aléatoires

5.3.1 Présentation générale de la méthode

Dans le contexte de l'extraction non supervisée de motifs temporels, et plus précisément de la fouille de caractères, Chiu and Keogh (2003) ont expérimenté un algorithme efficace fondé sur la comparaison des projections aléatoires, sur une dimension inférieure, des sous-séquences d'une séquence étudiée. Initialement proposée par Buhler and Tompa (2002) pour l'extraction de motifs à partir de séquences de nucléotides, il a ensuite été adapté pour l'étude de séries temporelles en ne s'intéressant qu'aux séries temporelles à une dimension. Duchêne (2004) l'a ensuite adaptée aux données hétérogènes et multidimensionnelles pour l'extraction de motifs sur des données recueillies dans le cadre de télésurveillance médicale à domicile (ou tout au moins sa simulation). Cette méthode présente en outre l'avantage d'être non supervisée et peut donc s'adapter à différents types de données sans nécessité une étude préalable de l'expert pour orienter la recherche. Elle fournit à partir d'une séquence temporelle d'une manière assez rapide les motifs les plus fréquents dans la séquence et leur position de manière approximative.

Dans la méthode telle que l'ont mis en oeuvre Chiu and Keogh (2003), des sous-séquences discrètes, dites sous-séquences de base, sont extraites par une fenêtre glissante de longueur fixe sur la série temporelle initiale. A chaque position de la fenêtre glissante, la sous-séquence discrète associée est calculée à partir des valeurs moyennes observées sur chaque segment de discrétisation composant la fenêtre, selon un ensemble de symboles équiprobables. L'opération de projection, qui donne son nom à la méthode, consiste à appliquer de manière aléatoire un masque de sélection de dimension prédéfinie aux sous-séquences ainsi identifiées : on obtient ainsi des sous-séquences projetées. On calcul ensuite une matrice de collisions entre les sous-séquences projetées qui nous permet d'avoir un indicateur de probabilité de la similarité des sous-séquences. C'est en cela que la méthode donne des résultats qu'on peut qualifier d'approximatif puisqu'elle ne se base pas sur une comparaison exhaustive des séquences deux à deux mais sur des extraits de ces séquences obtenus grâce aux masques de projections. On peut comparer le principe au mécanisme de reconnaissance visuelle qu'on peut mettre en oeuvre quand on compare deux scènes ou deux images. Sans avoir un niveau de détail très élevé (images floues ou scènes éloignées), on peut déjà présumer de la similarité entre les deux scènes avec un indice de confiance assez fort. L'intérêt de cet « à peu près » est qu'on peut ainsi étudier des séquences légèrement bruitées dans le cadre d'étude de données réelles, ou, dans notre cas, de masquer certaines caractéristiques pour mieux révéler la similarité entre d'autres.

5.3.2 Étapes du traitement

Deux choix se présentent à nous pour la mise en oeuvre de la méthode. Le premier consiste à utiliser la méthode originelle des projections aléatoires mise en oeuvre par Chiu and Keogh (2003) et qui s'applique à des données mono-dimensionnelles. En effet, les différents traitements présentés dans la partie 5.2 nous permettent d'abstraire suffisamment nos données pour les ramener à une simple série mono-dimensionnelle de symboles dans le temps (discrétisation >

réduction > agrégation (méthode simple)). L'autre approche consiste à utiliser la version adaptée par Duchêne (2004) pour le traitement de données hétérogènes et multi-dimensionnelles. Cette méthode ne nous dispense pas de la phase d'abstraction, mais la réduction ne sera pas effectuée (discretisation > agrégation (méthode évoluée)). Qu'on applique la première ou la deuxième méthode, les étapes restent les mêmes, seul certains points que nous allons présenter changent.

Résumons ci-dessous les étapes à effectuer pour appliquer cette méthode aux données :

1. **Abstraction des données :** À partir d'une séquence temporelle $\mathbb{X} = \{X_1, X_2, \dots, X_N\} | X_i = \{x_{1X_i}, x_{2X_i}, \dots, x_{nX_i}, q_{1X_i}, q_{2X_i}, \dots, q_{mX_i}\}$, un certain nombre de pré-traitement permettent d'obtenir une séquence simplifiée \mathbb{X}' adaptée à l'étude en terme de complexité et d'homogénéité des données.
2. **Projections aléatoires :**
 - (a) **Extraction des sous-séquences :** Les sous-séquences sont extraites par une fenêtre glissante par pas de 1 de longueur ω sur la séquence \mathbb{X}' . On obtient donc $N - \omega + 1$ sous-séquences.
 - (b) **Masquage des sous-séquences :** Un masque de projection est défini de manière aléatoire sur chaque sous-séquence. Les dimensions du masque sont fonctions de la dimension dans laquelle on a choisi de travailler (données mono ou multi-dimensionnelles).
 - **Données mono-dimensionnelle :** Masque de dimension ω_{masque} où $0 \leq \omega_{masque} < \omega$, ω_{masque} étant le nombre de symboles (ou vecteurs) masqués.
 - **Données multi-dimensionnelle :** Masque de dimension $\omega_{masque} \times p_{masque}$ où $0 \leq \omega_{masque} < \omega$ et $0 \leq p_{masque} < n + m$ où $p + m$ représente la taille d'un vecteur X_i de \mathbb{X} , p_{mask} étant le nombre de paramètres masqués dans chaque vecteur.
 - (c) **Matrice de collisions :** La matrice de collision M_{col} de dimension $(N - \omega + 1) \times (N - \omega + 1)$ entre les sous-séquences projetées (*i.e.* entre les séquences de valeurs non masquées) est mise à jour. Si deux sous-séquences i et j sont identiques, on incrémente la case leur correspondant : $M_{col}[i, j] = M_{col}[i, j] + 1$.
 - (d) **Seuil de collisions :** On réitère les opérations 2b et 2c pour que le nombre de collisions devienne significatif.

La plupart de ces opérations sont paramétrées. L'efficacité de cette procédure dépend donc des paramètres suivants :

- ω : nombre de symboles par sous-séquence,
- ω_{proj} : nombre de symboles projetés ($\omega - \omega_{masque}$),
- p_{proj} : nombre de paramètres projetés ($p - p_{masque}$),
- n_{proj} : nombre de projections (itérations de la procédure).

On distingue les paramètres *méthodologiques* dont on peut définir a priori une valeur appropriée dans un contexte donné (nombre de symboles par sous-séquence, nombre de symboles projetés, nombre de paramètres projetés), des paramètres *de réglages* correspondant aux différents seuils à adaptés en fonction des caractéristiques des données (seuil minimum de collision conditionnant n_{proj}).

La mesure servant de base à la comptabilisation des collisions est une mesure exacte, c'est à dire qu'on considérera deux sous-séquences projetées comme identique si elles sont totalement identiques : égalité entre chacun des paramètres de chacun des vecteurs de la sous-séquence.

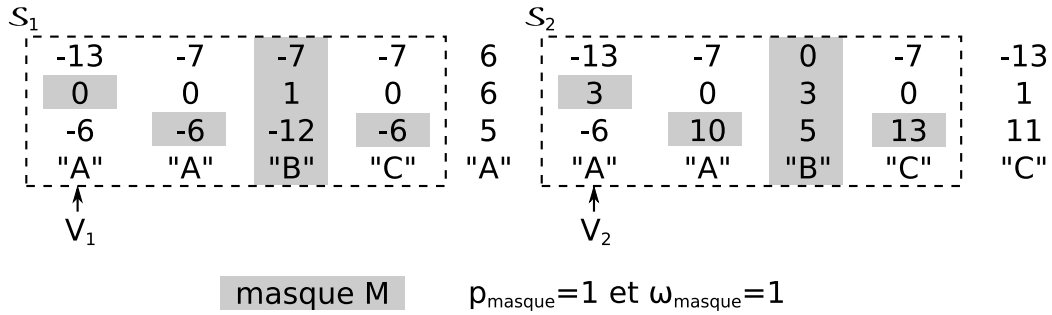


FIG. 5.4 – Principe d'utilisation des projections aléatoires dans Duchêne (2004) pour la tolérance au bruit.

5.3.3 Mise en oeuvre de la méthode

Nous ne détaillerons pas ici les procédures de réduction des données précédemment expliquées. Nous nous contenterons de détailler la façon dont nous avons développé la méthode des projections aléatoires pour l'étude de nos données de simulation. Comme précédemment, l'ensemble des implémentations dans le langage R des algorithmes décrits ci-après est disponible en annexe.

Choix préliminaires

La méthode des projections aléatoires permet de trouver les sous-séquences les plus fréquentes. Dans son travail, Duchêne (2004) utilise la méthode des projections aléatoires afin de caractériser des activités ordinaires de personnes sur une journée. Une activité est alors vue comme un enchaînement d'événements (4 dans les travaux en question). Elle cherche donc via cette méthode les séquences de 4 événements les plus fréquentes. Le contexte étant l'étude de données réelles issues de capteurs, les masques de projections permettent lors de la comparaison de sous-séquences pour le remplissage de la matrice de collisions de tolérer certaines différences tant au niveau des valeurs des vecteurs composant les sous-séquences, qu'au niveau de la séquence de vecteurs composant la sous-séquence. La tolérance au niveau des valeurs est liée au masquage de certaines valeurs par le masque de projection (paramètre p_{masque}). Ainsi, si on considère l'exemple de la figure 5.4, les vecteurs V_1 et V_2 sont considérés comme identiques selon le masque M alors que les valeurs respectives de leur deuxième attribut sont différentes. La tolérance au niveau de la séquence de vecteurs composant la sous-séquence est liée au masquage de certains vecteurs par le masque de projection (paramètre ω_{masque}). Ainsi, si on considère l'exemple de la figure 5.4, les sous-séquences S_1 et S_2 sont considérées comme identiques selon le masque M alors que leur troisième vecteur est différent. L'utilisation de la méthode des projections aléatoires permet ainsi d'être moins sensible au bruit présent dans toute donnée réelle. La matrice de collision traduit la fréquence de sous-séquences projetées, mais au final il s'agit bien des sous-séquences originelles dont on détermine la fréquence.

Dans notre cas, nous ne recherchons pas les activités les plus fréquentes, mais des succes-

sions d'évènements directement consécutifs ou non. Par *directement consécutifs*, on entend que deux évènements A et B ne sont pas successifs si n évènements intermédiaires sont intercalés entre A et B (ex. $ACFFEB$). Le motif correspondant est alors $A****B$. $*$ représente un évènement quelconque survenu entre A et B . Comme il est d'usage, nous le nommerons *joker*. Ainsi, ce n'est pas la fréquence des sous-séquences originelles qui nous intéresse, mais celle de sous-motifs contenus dans ces séquences et représentatif d'un enchaînement d'évènements plus ou moins lâche temporellement. De plus, nous ne travaillons par sur des données réelles mais sur des données issues de la simulation donc beaucoup moins sujettes au bruit. Le masque de projection n'assure donc pas tout à fait le même rôle que pour Duchêne (2004). Le masquage des paramètres permet toujours une tolérance entre dans la comparaison des sous-séquences. Le masquage des vecteurs assure lui non plus un rôle de tolérance, mais insère un joker dans une suite d'évènements de la sous-séquence. De cette façon, on pourra mettre en évident dans deux sous-séquences originelles différentes des successions d'évènements identiques correspondant à deux sous-séquences projetées identiques. Afin de détecter l'exhaustivité des successions de motifs possible pour une largeur de séquence donnée, nous allons donc générer la combinaison suivant la dimension temporelle de l'ensemble des masques possibles pour une largeur de séquence donnée, indépendamment donc de la dimension des attributs des vecteurs. Pour cette dernière dimension, deux choix s'offrent à nous. Le premier consiste à la conserver et adopter la même approche que Duchêne (2004) au niveau du masquage des attributs. Dans ce cas, pour chaque masque possible selon la dimension temporelle, on itérera n fois pour atteindre le seuil de collisions choisi. Le deuxième consiste à travailler directement sur une séquence abstraite mono-dimensionnelle, la tolérance de similarité entre deux vecteurs étant alors définie au niveau de la réduction des données. L'itération pour atteindre un seuil de collision n'est donc plus nécessaire puisque l'aspect aléatoire de la génération des masques disparaissant, n itération du traitement correspondrait alors à multiplier la matrice de collisions obtenu par la première itération par n . Le premier choix est plus coûteux en temps de traitement, mais présente le bénéfice de conserver l'intégrité dimensionnelle des vecteurs et l'esprit des projections aléatoires.

Voilà donc en synthèse les étapes du traitement que nous allons effectuer :

1. extraire l'ensemble des sous-séquences de la séquence temporelle d'origine,
2. générer l'ensemble des masques possibles pour la largeur de sous-séquence définie,
3. calculer pour chacun des masques la matrice de collisions,
4. déterminer en réassociant la matrice aux sous-séquences les successions d'évènements fréquentes.

Intéressons nous rapidement à la combinatoire de ces traitements :

- **La génération des sous-séquences** : Pour une séquence temporelle de longueur N et des sous-séquences de longueur ω , on trouve $N - \omega + 1$ sous-séquences possibles.
- **La génération des masques de projections** : Pour des sous-séquences de longueur ω , pour une valeur ω_{proj} donnée, il existe $C_{\omega_{proj}}^{\omega}$ combinaisons de masque possibles. Les valeurs d' ω_{proj} égales à 0 et 1 étant vides de sens (on obtient respectivement une projection vide et une projection d'un seul symbole), l'ensemble des masques possibles pour une sous-séquence donnée est donc égale à $\sum_{i=2}^{\omega_{proj}} C_i^{\omega}$. À titre illustratif, pour $\omega = 4$, on obtient 11 masques possibles.

Pour effectuer cette recherche, il est essentiel de définir un certain nombre de paramètres clés. Ces paramètres correspondent à ceux que nous avons présentés dans la section 5.3.2. Ils revêtent néanmoins une signification particulière dans le contexte de notre étude.

Le premier paramètre est la largeur ω des sous-séquences. Dans le cas de Duchêne (2004) a considéré que l'activité d'une personne pouvait se réduire au minimum à une séquence de 4 motifs. Sa recherche est donc basée sur une fenêtre temporelle d'une largeur correspondante de 4. L'intervalle de temps exploré est donc relativement petit et la combinatoire associée également (le masque utilisé est paramétré avec ω_{proj} égal à 3 (1 motif de masqué par séquence), on obtient $C_3^4 = 4$). Notre approche s'en différencie selon 2 points principaux. D'une part, il ne s'agit pas pour nous de caractériser des activités mais des liens de causalité entre une suite d'événements. L'intervalle de temps exploré est donc variable. Il est conditionné d'une part par la granularité temporelle des événements traités et par le choix de l'expérimentateur sur la largeur de la plage de variation qu'il souhaite explorer. D'autre part, dans sa méthode, Duchêne (2004) réassocie les séquences originelles (non projetées) aux similarités détectées. En effet, les projections aléatoires lui servent à introduire une certaine approximation dans la mesure de similarité entre sous-séquences et ainsi d'être moins sensible au bruit, fréquent pour les données réelles qu'elle étudie. Ainsi, les motifs recherchés ne sont pas les séquences projetées mais bien les séquences originelles. Dans notre cas, les séquences projetées sont les motifs recherchés car elles correspondent à un enchaînement non successif d'événements.

Le choix des paramètres du masque de projection (à savoir ω_{proj} et p_{proj}) conditionne le type de causalité temporelle que l'on désire trouver. Plus ω_{proj} est important plus le nombre de symboles projetés est important et donc plus les motifs fréquents trouvés comporteront d'événements. Le rapport $\frac{\omega_{proj}}{\omega}$ indique le rapport entre la taille des motifs recherchés par rapport à la taille globale des sous-séquences extraites. La combinatoire du nombre de masques différents si on se place dans le contexte mono-dimensionnel ($p = 1$) est $C_{\omega}^{\omega_{proj}}$. Pour exemple, des masques masquant 5 motifs sur des sous-séquences en contenant 10, nous donnent 252 masques possibles. Dans un contexte multi-dimensionnel, on obtient $C_{\omega}^{\omega_{proj}} \times C_p^{p_{proj}}$. Pour le même exemple que précédemment, avec cette fois-ci des vecteurs de 10 paramètres dont 3 sont masqués, cela nous donne 14112 masques possibles.

Nous avons vu dans la partie 5.2 que la phase de pré-traitement pouvait nous fournir aux choix des données multi-dimensionnelles (*attributs* \times *temps*) ou mono-dimensionnelles (*temps* uniquement) après une phase de réduction. Considérons donc ces deux dimensions en regard de la méthode des projections aléatoires, et notamment du masque de projection. Dans un contexte multi-dimensionnel, le masque des projections joue un rôle différent pour chaque dimension. Pour la dimension des attributs, le masquage de certaines valeurs de vecteurs en fonction du paramètre p_{masque} permet de tolérer un certain pourcentage de différence entre des vecteurs de sous-séquences lors du remplissage de la matrice de collisions. L'avantage est qu'on peut ainsi rapprocher deux événements qui ne sont pas strictement identiques. Pour la dimension temporelle, le masquage de vecteurs permet de trouver des causalités entre des événements non directement successifs.

Notre méthode consiste donc, pour l'ensemble de taille $N - \omega + 1$ des sous-séquences de taille ω , à générer l'ensemble des projections possibles de ces sous-séquences. Soit, si l'on

reprend les formules données précédemment, $\sum_{n=1}^{\omega} C_n^{\omega}$. On itère pour générer l'ensemble des masques de projections, appliqués à l'ensemble des sous-séquences extraites.

La génération des sous-séquences

Afin de générer les sous-séquences, on va parcourir la séquence temporelle principale avec une fenêtre glissante de dimension ω et en extraire l'ensemble des sous-séquences de cette taille. Il s'agit donc d'une simple boucle de parcours décrite dans l'algorithme 5.7. Le choix de la valeur de ω conditionne la taille de la fenêtre sur laquelle on va rechercher des causalités temporelles. Ce choix est donc déterminant suivant que l'on cherche à caractériser plutôt des causalités d'évènements sur du court, du moyen ou du long terme.

Algorithme 5.7 Génération des sous-séquences.

```

1: Fonction GENERER-SOUS-SEQUENCES( $\mathbb{X}$ ,  $\omega$ )
2:   ▷  $\mathbb{X}$  : la série temporelle de taille  $N$  dont on veut extraire les sous-séquences
3:   ▷  $\omega$  : la taille des sous-séquences extraites
4:   ▷ Retourne  $V_{ss-seq}$  : un vecteur des sous-séquences
5:    $V_{ss-seq} \leftarrow$  nouveau vecteur de taille  $N - \omega + 1$ 
6:   Pour  $i$  allant de 1 à  $N - \omega + 1$  faire
7:      $V_{ss-seq}[i] \leftarrow$  SOUS-ENSEMBLE( $i$ ,  $i + \omega - 1$ )
8:   Fin Pour
9:   Retourner  $V_{ss-seq}$ 
10: Fin Fonction

```

Génération du masque et projection des sous-séquences

La génération du masque en elle-même n'est pas une étape très complexe et l'application du masque généré à une sous-séquence pour en effectuer la projection est également quelque chose d'assez simple. Le masque est généré sous la forme d'une matrice de taille $\omega \times p$ où les 1 symbolisent les attributs masqués (ou supprimés), les 0 ceux qui sont conservés. La projection consiste ensuite essentiellement à parcourir ce masque et effectuer l'équivalent d'un *ou exclusif* sur la matrice *vecteurs* \times *temps* représentant la séquence.

Le point le plus sensible est la génération de l'exhaustivité des masques en fonction du type de motif recherché. Si nous prenons par exemple des sous-séquences de taille 4, plusieurs motifs peuvent alors être trouvés :

- 2 évènements séquentiels comportant de 0 à 2 évènements intermédiaires (XX , $X*X$ et $X**X$ ⁷⁷,
- 3 évènements séquentiels comportant de 0 à 1 évènement intermédiaire (XXX , $XX*X$ ou $X*XX$),
- 4 évènements séquentiels ($XXXX$).

⁷⁷* est ici utilisé comme métacaractère symbolisant les évènements intermédiaires.

Cela correspond respectivement aux masques avec le paramétrage suivant :

- $\omega_{proj} = 2$ pour XX^{**} , $*XX^{*}$, $**XX$, $X^{*}X^{*}$, $*X^{*}X$ et $X^{**}X$,
- $\omega_{proj} = 3$ pour $*XXX$, XXX^{*} , $XX^{*}X$ ou $X^{*}XX$,
- $\omega_{proj} = 4$ pour $XXXX$ ⁷⁸.

Pour une sous-séquence de taille 4, l'ensemble des motifs de taille inférieure ou égale à 4 seront trouvés en générant l'ensemble des masques possibles pour ω_{proj} variant de 2 à 4. De manière plus générale, on peut montrer que, pour une séquence de taille ω , la génération de l'ensemble des masques de projections pour un ω_{proj} variant de 2 à $\omega - 1$ est suffisante pour extraire tous les motifs sur cette chaîne et ses sous-chaînes (motifs de taille 2 à 4 donc). Ce qui nous donne donc, comme indiqué dans la section 5.3.3, $\sum_{i=2}^{\omega-1} C_i^{\omega}$ masques possibles (10 avec $\omega = 4$).

Nous avons choisi d'exclure les masques comportant des métacaractères (*) à leurs extrémités dans la génération de nos masques. En effet, si l'utilisateur choisi d'effectuer une recherche sur des sous-séquences de taille 4, les masques XX^{**} , $*XX^{*}$ et $**XX$ sont équivalents à une recherche des sous-séquences similaires de taille 2. Il est donc plus pertinent dans ce cas de directement effectuer sa recherche sur les sous-séquences de cette taille, d'autant plus que les 3 masques précédemment cités vont d'une part augmenter artificiellement le taux de collision des sous-séquences ne se situant pas aux extrémités de la série temporelle, et d'autre part augmenter inutilement la combinatoire de recherche par rapport à une recherche directe sur les sous-séquences de taille 2. Ainsi, pour une sous-séquence de taille 4, nous ne générerons que les masques suivants : $X^{**}X$, $XX^{*}X$ et $X^{*}XX$ ⁷⁹. Cela nous donne donc un nombre de masques égal à : $\sum_{i=1}^{\omega-2} C_i^{\omega-2}$. L'algorithme 5.8 détaille la méthode employée pour la génération des masques et la projection des sous-séquences.

Remplissage des matrices de collisions

Dans la méthode originale des projections aléatoires, une seule matrice de collisions stocke la synthèse des collisions obtenues pour l'ensemble des masques appliquées aux sous-séquences. L'objectif est en effet de comptabiliser la similarité entre des sous-séquences a priori différentes mais rendues identiques par l'application d'un ou plusieurs masques. Chacun de ces cas d'identité contribue à augmenter le nombre de collisions entre ces sous-séquences jusqu'à atteindre le seuil fixé au départ par l'utilisateur.

Dans notre cas, nous ne souhaitons pas agréger la contribution de masques différents à la similarité entre deux sous-séquences, mais au contraire comptabiliser pour chaque masque les similarités entre sous séquences masquées ainsi générées. Ce n'est pas la similarité entre sous-séquences qui nous intéresse mais celle entre des séquences de motifs comportant des métacaractères, ces dernières étant générées par des masques. Par exemple, nous construisons la matrice de collision pour toutes les séquences de motifs du type $X^{*}XX$. Nous n'avons donc

⁷⁸Ce paramétrage ne constitue pas réellement un masque de projection car aucun attributs ne sera masqué. La séquence restera donc inchangée

⁷⁹Le cas $XXXX$ n'étant pas une véritable projection, il sera traité par la simple recopie du vecteur des sous-séquences. Ceci explique la dimension $\sum_{i=1}^{\omega-2} C_i^{\omega-2} + 1$ au lieu de $\sum_{i=1}^{\omega-2} C_i^{\omega-2}$ du vecteur de résultats dans l'algorithme 5.8.

Algorithme 5.8 Projection des sous-séquences.

```

1: Fonction PROJECTIONS( $V_{ss-seq}$ )
2:   ▷  $V_{ss-seq}$  : un vecteur de sous-séquences de taille  $\omega \times p$ 
3:   ▷ Retourne  $V_{ss-seq-projetees}$  : le vecteur contenant autant de vecteurs de sous-séquences
   temporelles qu'il y a de masques à générer et à appliquer
4:    $V_{ss-seq-projetees} \leftarrow$  nouveau vecteur de vecteurs de taille  $\sum_{i=1}^{\omega-2} C_i^{\omega-2} + 1$ 
5:    $compteur_{masques} \leftarrow 0$ 
6:   Pour  $i$  allant de 1 à  $\omega - 2$  faire
7:      $masques \leftarrow$  GENERER-MASQUES( $i$ )
8:     Pour  $j$  allant de 1 à  $C_i^{\omega-2}$  faire
9:        $masque_{courant} \leftarrow masques[j]$ 
10:       $compteur_{masques} \leftarrow compteur_{masques} + 1$ 
11:      Pour  $k$  allant de 1 à taille de  $V_{ss-seq}$  faire
12:         $V_{ss-seq-projetees}[compteur_{masques}, k] \leftarrow$ 
           APPLIQUER-MASQUE( $V_{ss-seq}[k], masque_{courant}$ )
13:      Fin Pour
14:    Fin Pour
15:  Fin Pour
16:   $V_{ss-seq-projetees}[compteur_{masques} + 1] \leftarrow V_{ss-seq}$ 
17:  Retourner  $V_{ss-seq-projetees}$ 
18: Fin Fonction

19: Fonction GENERER-MASQUES( $\omega_{proj}$ )
20:   ▷ Fonction générant l'ensemble des masques pour une valeur de  $\omega_{proj}$  donnée
21: Fin Fonction

22: Fonction APPLIQUER-MASQUE(sequence, masque)
23:   ▷ Fonction permettant d'appliquer un masque de projection à une sous-séquence
24: Fin Fonction

```

pas une seule matrice de collision mais autant qu'il y a de masques possibles pour la taille de sous-séquence (ω) sur laquelle nous travaillons.

Algorithme 5.9 Remplissage des matrices de collisions.

```

1: Fonction REMPLISSAGE-MATRICES-COLLISIONS( $V_{ss-seq-projetees}$ )
2:   ▷  $V_{ss-seq-projetees}$  : un vecteur de sous-séquences projetées
3:   ▷ Retourne  $V_{matrices-collisions}$  : le vecteur des matrices de collisions entre les sous-
      séquences
4:    $V_{matrices-collisions} \Leftarrow$  nouveau vecteur de matrices de taille  $\sum_{i=1}^{\omega-2} C_i^{\omega-2} + 1$ 
5:   Pour  $i$  allant de 1 à taille de  $V_{ss-seq-projetees}$  faire
6:     Pour  $j$  allant de 1 à taille de  $V_{ss-seq-projetees}[i] - 1$  faire
7:       Pour  $k$  allant de  $j + 1$  à taille de  $V_{ss-seq-projetees}[i]$  faire
8:         Si  $V_{ss-seq-projetees}[i, j] = V_{ss-seq-projetees}[i, k]$  alors
9:            $V_{matrices-collisions}[i][j, k] \Leftarrow 1$ 
10:           $V_{matrices-collisions}[i][j, k] \Leftarrow V_{matrices-collisions}[i][k, j]$ 
11:         Fin Si
12:       Fin Pour
13:     Fin Pour
14:   Fin Pour
15:   Retourner  $V_{matrices-collisions}$ 
16: Fin Fonction

```

Les matrices de collisions ainsi calculées nous permettent in fine d'obtenir les séquences de motifs bénéficiant du plus fort support afin de les présenter à l'utilisateur. Un filtrage via un seuil minimum de représentativité peut être appliqué pour diminuer le nombre de résultats présentés et ne conserver que les plus pertinents.

Conclusion

Dans cette partie, nous avons décrit les algorithmes et méthodes de fouille de données que nous avons développés pour traiter le cas spécifique de nos données de simulation. En nous basant sur les précédents travaux sur la méthode des projections aléatoires, nous avons pu mettre en place une méthode d'ECD, pertinente dans le contexte qui nous intéresse, du prétraitement à la restitution des résultats. Nous avons ainsi présenté notre méthode d'abstraction des données afin de les rendre exploitables par la méthode des projections aléatoires, profitant dans cette première phase de la qualité des données de simulation par rapport aux données réelles nous dispensant de la plupart des étapes classiques de traitement du bruit et des données manquantes. Nous avons ensuite détaillé notre adaptation de la méthode des projections aléatoires, initialement destinée à détecter des motifs dans des données séquentielles avec une tolérance aux différences, pour dans notre cas trouver des séquences d'évènements déterminées entre lesquels peuvent s'intercaler des évènements non déterminées. Nous réalisons ainsi une fouille sur des séquence d'évènement non systématiquement consécutifs, on parle aussi de chroniques

temporelles.

6

Implémentation de la simulation et résultats obtenus

Sommaire

6.1	Implémentation de la plate-forme de simulation	152
6.1.1	Le langage Java	152
6.1.2	La plate-forme SMA JADE	152
6.1.3	Le logiciel R	153
6.2	Déroulement de la simulation	154
6.3	Fouille des traces d'agent	156

Introduction

Nous avons déjà vu dans les deux chapitres précédents d'une part les modèles mis en place pour construire notre agent émotionnel et à plus une grande échelle la simulation multi-agents entière, et d'autre part les outils de fouille de données spécifiquement élaborés pour le traitement des données produites par notre simulation. Nous allons maintenant présenter la mise en œuvre de ces éléments et voir comment du déroulement d'une simulation, nous pouvons extraire des connaissances pertinentes.

Nous présenterons d'abord les aspects techniques et les choix d'implémentation que nous avons effectués pour concevoir nos outils. Nous présenterons ainsi les langages et plates-formes de conception utilisés à cette fin. Dans une deuxième partie, nous détaillerons le déroulement d'une simulation en présentant l'interface de la simulation et en mettant en évidence les faits qui y sont déjà observables. Enfin, nous présenterons les résultats des algorithmes de fouille sur les données produites.

6.1 Implémentation de la plate-forme de simulation

6.1.1 Le langage Java

Le langage Java est aujourd'hui largement connu et plébiscité par la communauté informatique. Le but ici n'est donc pas de revenir sur ce qu'est Java en en faisant une présentation générale, mais plutôt de voir en quoi il est particulièrement adapté aux développements de systèmes multi-agents. En premier lieu, Java est un langage objet. Les agents pouvant être considérés comme des objets avancés, ce type de langage est le mieux adapté pour les développer. De plus, la modularité apportée par les langages objet devient rapidement indispensable dès lors qu'on modélise des agents dont l'architecture est un tant soit peu complexe. Par ailleurs, un des principes fondamentaux des systèmes multi-agents est l'hétérogénéité des architectures et des technologies pouvant être employées pour le développement et l'exécution des agents. Java est un langage semi-interprété qui est exécuté par une machine virtuelle (la JVM⁸⁰ déclinée sur de nombreuses plates-formes (Linux, AIX, Mac OS, Solaris, Windows, ...)). La portabilité du code que cela apporte permet ainsi de déployer un système multi-agents à travers un réseau hétérogène de machines. D'autre part, Java est un langage construit pour le Web. Un des premiers objectifs était de rendre disponible via le mécanisme d'*applet* un applicatif lourd via le client léger qu'est le navigateur Web. On trouve ainsi le couple de protocoles IIOP et RMI⁸¹, basés sur la norme CORBA⁸², permettant l'invocation de méthodes à distance. Ce système est utilisé dans de nombreux outils multi-agents tel que JADE que nous présentons ci-dessous.

6.1.2 La plate-forme SMA JADE

La plate-forme JADE⁸³ est un outil développé par CSELT⁸⁴ en Java fournissant des bibliothèques de classes pour le développement d'agents et permettant d'animer ces agents au sein d'une plate-forme SMA. Cet outil est utilisé par une grande partie de la communauté multi-agents et prend en charge certaines caractéristiques essentielles aux multi-agents telles que la communication et la concurrence sans pour autant contraindre exagérément la structure de l'agent. C'est d'après le conseil de chercheurs, des comparatifs entre plates-formes (Garneau and Delisle, 2002) et notre propre étude comparative que nous avons choisi cet outil qui a de plus l'avantage d'être gratuit. Très utilisé, il est aussi abondamment documenté et de nombreux exemples sont disponibles. JADE supporte une distribution sur plusieurs machines ce qui peut-être intéressant pour répartir les coûts de calcul parfois lourd des SMA. Il dispose d'une interface permettant le contrôle et la supervision des agents. Enfin, et c'est ce qui nous intéresse ici, il est compatible FIPA-ACL et propose un système de gestion d'agents, un facilitateur

⁸⁰Pour *Java Virtual Machine*.

⁸¹IIOP signifie *Internet Inter-Orb Protocol* et RMI, *Remote Method Invocation*.

⁸²Pour *Common Object Request Broker Architecture*, standard de l'OMG définissant l'architecture logicielle nécessaire pour permettre à des parties de programmes (objets) de communiquer avec d'autres, issues d'environnements différents.

⁸³Pour *Java Agent DEvelopment framework*.

⁸⁴Pour *Centro Studi E Laboratori Telecomunicazioni*, laboratoire de Telecom Italia (Tilab).

d'annuaire et un canal de communication entre agents⁸⁵.

L'ensemble de la plate-forme JADE est elle-même un SMA. Le système de gestion d'agents est l'agent qui exerce le contrôle de supervision sur l'accès et l'usage de la plate-forme. Il est responsable de l'authentification des agents résidents et du contrôle des enregistrements. Le canal de communication entre agents est l'agent qui fournit le canal pour les interactions de base entre les agents. Le facilitateur d'annuaire est l'agent qui fournit un service de pages jaunes à la plate-forme multi-agents (Florea et al., 2002). Chacun de ces services rend transparente une bonne partie de la gestion de la communication entre agents tout en laissant de nombreuses libertés pour l'expression des messages.

6.1.3 Le logiciel R

Nous avons mené nos expérimentations avec le logiciel R (R Development Core Team, 2005) que nous allons présenter maintenant. R a été initialement créé par Robert Gentleman et Ross Ihaka du département de statistique de l'Université d'Auckland en Nouvelle Zélande. Depuis 1997, c'est une équipe (la «R Core Team») qui développe R. C'est à la fois un langage et logiciel qui permet de réaliser des analyses statistiques.

R possède :

- un système efficace de manipulation et de stockage des données,
- différents opérateurs pour le calcul sur tableaux,
- un grand nombre d'outils pour l'analyse des données et les méthodes statistiques,
- des moyens graphiques pour visualiser les analyses,
- un langage de programmation simple et performant comportant : conditions, boucles, moyens d'entrées sorties, possibilité de définir des fonctions récursives.

Le langage R s'est construit à partir de deux autres langages :

- S qui est un langage développé par les AT&T Bell Laboratories. S est à la fois un langage haut niveau et un environnement pour l'analyse des données et les représentations graphiques. Il est utilisable à travers le logiciel SPlus qui est commercialisé par la société Insightful. C'est l'un des logiciels de statistiques les plus populaires et il s'est imposé comme une référence dans le milieu statistique.
- Scheme de Sussman qui est un langage fonctionnel dont le principe fondamental est la récursivité. L'exécution et la sémantique de R en sont dérivées. Le noyau de R est écrit en langage machine interprété qui a une syntaxe similaire au langage C, mais qui est réellement un langage de programmation avec des capacités identiques au langage Scheme.

La plupart des fonctions accessibles par l'utilisateur dans R sont écrites en R. Pour les tâches plus coûteuses les langages C, C++ et Fortran ont été utilisés et liés pour une meilleure efficacité. L'utilisateur peut créer de nouvelles fonctions en R ou en C pour manipuler directement des objets R. Toutes les grandes méthodes statistiques classiques sont implémentées dans le logiciel

⁸⁵Respectivement, *Agent Management System (AMS)*, *Directory Facilitator (DF)* et *Agent Communication Channel (ACC)*, en anglais.

R : les modèles linéaires, les modèles linéaires généralisés, la régression nonlinéaire, les séries chronologiques, les tests paramétriques et non paramétriques classiques, ... R peut facilement étendre ses fonctions par l'intermédiaire de bibliothèques. Les modules de bases sont fournis avec la distribution de R mais d'autres sont disponibles par l'intermédiaire du CRAN (Comprehensive R Archive Network). Le CRAN est un ensemble de sites qui fournissent ce qui est nécessaire à la distribution de R, ses extensions, sa documentation, ses fichiers sources et ses fichiers binaires. Les extensions ont été créées pour des buts spécifiques et proposent une large gamme de statistiques modernes : analyse descriptive des données multidimensionnelles, arbres de régression et de classification, graphiques en trois dimensions, ...

Enfin, R est développé pour pouvoir être utilisé avec les systèmes d'exploitation Unix, GNU/Linux, Windows et MacOS. C'est un logiciel libre qui est distribué sous les termes de la GNU Public Licence et il fait partie intégrante du projet GNU. Le logiciel étant donc dans le domaine public, son point fort est le développement d'applications, de modules qui sont mis à la disposition de tous les utilisateurs et développeurs qui peuvent proposer des extensions. R est donc en perpétuelle évolution et son potentiel est donc très grand.

6.2 Déroulement de la simulation

Dans cette partie, nous allons montrer l'interface de la simulation et expliquer son déroulement. La figure 6.1 présente une capture d'écran de la vue principale de la simulation sur laquelle on peut observer les agents s'affairant sur une chaîne de conditionnement. Nous y voyons la chaîne telle que décrite en section 4.2.1. Les bulles permettent de voir les communications entre agents s'effectuer. Les phrases choisies sont assez illustratives pour pouvoir distinguer l'intention de l'émetteur. Enfin le chef de ligne est là pour rassurer les employés de la chaîne dans les cas où des problèmes peuvent survenir. Il ne possède pas d'état psychologique et il n'a été implémenté que pour simuler le rôle positif des encadrants de la chaîne qui vient contrebalancer les aléas négatifs que peuvent rencontrer les employés.

La principale répercussion physique de l'état émotionnel était l'expression faciale d'une personne. C'est pourquoi nous avons choisi de représenter un agent par une simple figure. Le visage est décrit par la position des sourcils, des lèvres. Chaque état émotionnel correspond donc à une image (figure 6.2). Ainsi la tristesse se traduit par des sourcils vers le haut et des lèvres vers le bas. Aux descriptions proposées par Eckman, nous avons ajouté un autre paramètre qu'est le tempérament et plus particulièrement le niveau d'extraversion. En effet, une des principales caractéristiques de l'extraversion est de montrer plus facilement ce que l'on ressent. Nous avons choisi de mettre en valeur cette extraversion ou introversion en ouvrant ou fermant la bouche des agents. En effet, l'ouverture de la bouche donne l'impression d'une émotion plus forte ce qui correspond bien à notre cas. Si deux personnes ressentent de la joie à la même intensité, l'entourage pensera que la personne extravertie ressent plus de joie que celle qui est introvertie. Nous avons donc choisi de représenter nos agents dans l'esprit des smileys et sur une base psychologique. Ceci permet d'obtenir un dessin simple donc compréhensible par tous sans ambiguïté dans la distinction des états émotionnels. Cela correspond d'autre part aux mes-

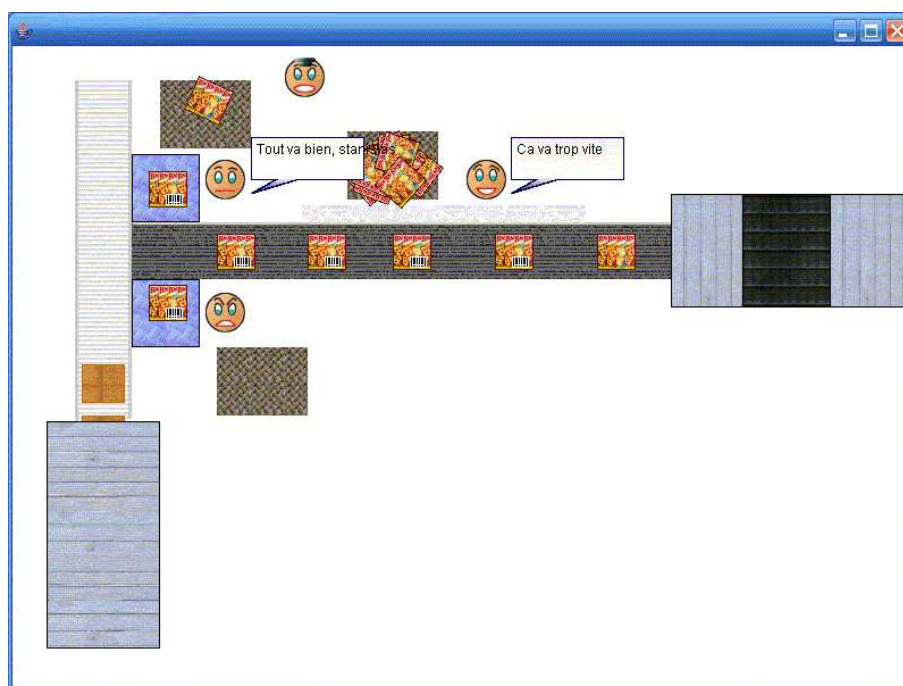


FIG. 6.1 – Capture d’écran de la fenêtre principale de la simulation.

sages transmis par l’agent sur son état émotionnel lors de ces interactions avec les autres entités du système.

motion temp rament	emotion				
	joie	tristesse	go t	d go t	col re
extraversion					
introversion					

FIG. 6.2 – Représentation des émotions.

Ensuite, chaque agent a une fenêtre pour son état interne (figure 6.3). Celle-ci permet d’avoir sous les yeux le tempérament de l’agent représenté sur un histogramme comme les rapports Dia-lecho. Dans cet exemple, l’agent est de type adaptation. Puis, nous pouvons choisir les variables dont nous voulons visualiser l’évolution. Ici, la joie, le goût, la gratitude, la quiétude, le stress et la fatigue mentale. Cette représentation en coordonnées parallèles est intéressante car elle permet de faire des liens entre variables par une simple visualisation (« la quiétude et le goût semble évoluer ensemble... »). Ce graphique permet d’envisager des grandes possibilités liées à un tel outil. En effet, on a, en perspective, de sauvegarder toutes les données issues de cette simulation et de mettre en valeur leurs particularités par des outils de visualisation en se servant des capacités cognitives humaines. Cette capture fournit une vue intéressante de l’évolution des

caractéristiques de l'agent, elle permet en effet de déceler l'apparition de motifs répétitifs. Elle justifie donc la fouille de données effectuée par la suite sur l'évolution de ces caractéristiques.

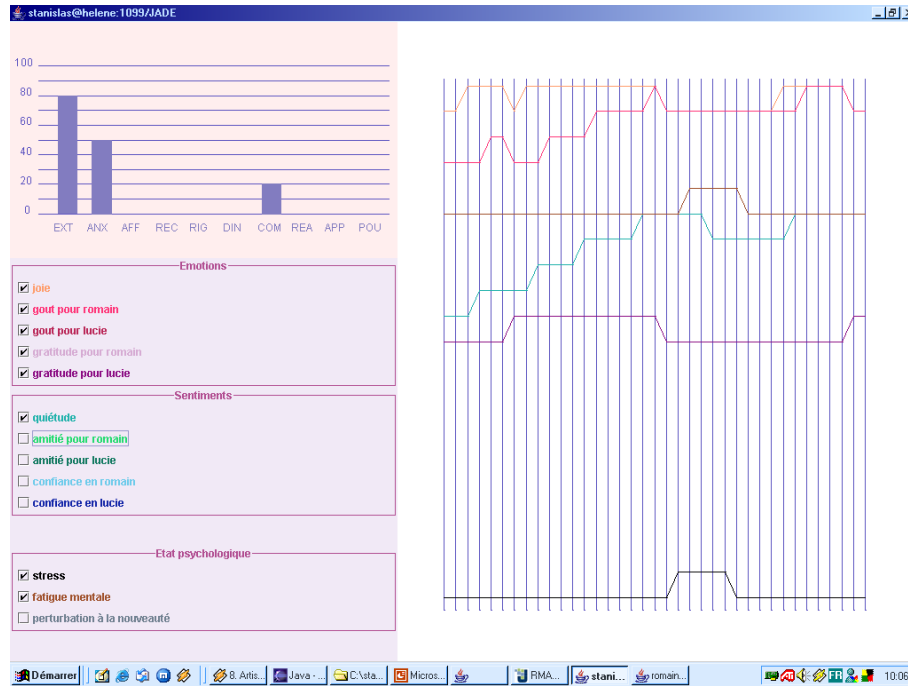


FIG. 6.3 – Évolution de l'état interne de l'agent.

6.3 Fouille des traces d'agent

Dans cette partie, nous allons nous limiter pour l'expérimentation de l'algorithme aux seules émotions ceci pour des raisons de présentation de résultats simples et lisibles. Nous avons à notre disposition un jeu de données de 500 itérations chronologiques issues de la simulation, contenant donc 500 fois le vecteur des 11 émotions pour un agent donné. Il nous a paru plus intéressant d'étudier les variations des émotions plutôt que leur valeur. En effet, le fait de savoir qu'une émotion passe souvent des valeurs 40, puis 60, puis 50, a peu d'intérêt, alors que découvrir que l'émotion augmente puis redescend de moitié aussitôt est porteur de plus d'information.

La réduction des données produit déjà des résultats visuellement interprétables. Pour rappel, cette étape consiste à transformer des vecteurs multidimensionnels en un symbole associé. Comme le montre la capture d'écran fournie par le logiciel R présentée en figure 6.4, nous pouvons déjà remarquer la répétition de certains vecteurs. Par exemple, les symboles V1 et V2 apparaissent plusieurs fois sur les 80 premières itérations. Ceci tend à indiquer qu'on retrouve des variations similaires entre deux instants successifs à plusieurs moments de la simulation. D'autre part, certains vecteurs, donc certaines variations, se retrouvent régulièrement tout au

[illegible]

FIG. 6.6 – Support des sous-séquences de taille 4 pour le masque XX^*X .

[1]	10	1	1	2	2	1	1	1	9	1	1	1	1	4	1	10	1	1	6	1	1	1	1	2	
[26]	1	1	10	1	2	6	5	1	3	1	1	10	1	1	2	3	1	1	8	1	12	2	1	6	1
[51]	6	10	1	1	8	1	1	1	1	10	1	1	1	8	1	1	5	1	1	1	2	1	1	6	3
[76]	8	4	3	3	3	5	22	2	3	2	2	2	1	5	2	1	6	2	8	1	4	3	1	1	1
[101]	1	4	5	6	1	6	10	1	3	6	2	1	2	1	1	1	1	1	1	3	3	1	2	10	1
[126]	2	1	1	2	2	1	1	3	3	2	1	1	10	8	1	3	8	1	1	4	4	2	5	1	10
[151]	6	1	2	1	4	1	5	1	2	1	1	1	1	1	4	1	1	5	5	1	4	6	3	1	1
[176]	3	2	1	1	6	1	9	5	1	2	6	5	1	4	2	2	6	6	12	1	8	2	1	1	1
[201]	1	1	1	2	1	1	6	1	1	8	1	1	5	2	2	1	3	6	1	8	12	8	3	1	22
[226]	1	1	4	10	6	3	2	4	4	4	4	8	1	2	2	1	2	1	2	6	1	3	1	2	10
[251]	1	3	1	22	6	2	8	3	1	1	2	22	2	1	22	3	1	22	6	1	22	1	2	10	1
[276]	12	3	3	1	3	2	1	2	2	2	1	6	1	1	4	1	1	1	1	2	8	2	1	2	1
[301]	3	1	2	22	1	6	12	9	12	2	4	8	4	3	22	1	4	4	4	2	6	2	1	22	2
[326]	1	22	5	3	22	4	8	6	8	10	2	12	9	3	4	22	2	8	10	1	22	4	1	12	5
[351]	22	4	9	10	8	12	9	6	8	12	8	6	5	22	1	4	2	1	4	4	1	8	6	4	10
[376]	8	6	8	12	8	4	8	10	8	2	8	22	2	8	22	2	9	12	4	1	8	2	1	22	1
[401]	4	6	8	2	8	3	3	22	1	9	10	9	10	1	22	1	3	22							

FIG. 6.7 – Support des sous-séquences de taille 4 pour le masque $X^{**}X$.

```
[ [4] ]  
[1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
[38] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
[75] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1  
[112] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
[149] 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
[186] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1 1  
[223] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1  
[260] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
[297] 1 1 1 1 1 1 1 1 1 2 2 2 1 1 1 1 1 1 1 1 1 1 1 1 1 3 3  
[334] 2 3 1 1 1 1 1 2 1 3 2 1 1 1 2 1 1 3 2 2 2 3 2 1 3 3 1 1  
[371] 1 1 1 1 3 2 2 2 1 1 1 1 2 1 1 1 1 2 2 1 1 1 1 1 2 2 1 1  
[408] 1 1 2 2 2 2 1 1 1 1
```

FIG. 6.8 – Support des sous-séquences de taille 4 pour le masque XXXX.

Conclusion

Nous avons, dans ce chapitre, détaillé la mise en œuvre d'une part des modèles d'agent et de SMA que nous avons présentés au chapitre 4, et d'autre part la réalisation de l'extraction des connaissances à partir des données de simulation conformément aux algorithmes et méthodes présentées dans la partie 5. Pour ce qui concerne la simulation multi-agents, nous avons indiqué nos choix d'implémentation et pourquoi la technologie Java et plus particulièrement la plate-forme JADE nous semblaient bien adaptées à nos besoins. Nous avons ensuite présenté le logiciel de simulation ainsi que le déroulement d'une simulation et les informations présentées à l'utilisateur. Pour ce qui concerne la fouille de données, nous avons tout d'abord présenté le logiciel libre R largement utilisé dans le milieu universitaire et particulièrement rapide et efficace pour effectuer des traitements statistiques et d'ECD. Nous avons enfin détaillé le travail de foule et la pertinence des connaissances restituées à l'utilisateur.

Conclusion

Synthèse

Dans ce mémoire de thèse, nous proposons une démarche originale pour l'étude des comportements humains qui couple une simulation comportementale basée sur les systèmes multi-agents à un processus d'extraction des connaissances à partir des données de simulation. Dans le cadre de la modélisation d'une entité virtuelle dotée d'un comportement affectif convaincant, nous nous appuyons sur le modèle d'agent cognitif BDI pour l'enrichir d'une dimension affective. Nous proposons ainsi un nouveau modèle d'agent émotionnel dénommé agent EFT. Du point de vue de l'extraction des connaissances à partir des données de simulation, nous adaptons la méthode des projections aléatoires utilisée pour la fouille de motifs fréquents dans les séquences temporelles afin de rechercher des occurrences de séquences d'évènements non consécutifs.

Pour clore ce manuscrit, nous rappelons les résultats obtenus et les apports de nos travaux de recherche (Daviet, 2006; Daviet et al., 2005a,b, 2008, 2006, 2005c; Fossé et al., 2006). Nous terminons enfin par les perspectives ouvertes par ces travaux.

Apports de nos travaux et résultats

EFT : un nouveau modèle d'agent émotionnel

Dans le cadre de la simulation du comportement humain, nous avons pu voir que les modèles d'agent émotionnel existants comportaient des lacunes notamment du fait d'une prise en compte de la dimension affective de l'individu réduite aux seules émotions. Nous proposons un nouveau modèle d'agent comprenant à la fois la modélisation de l'architecture et des processus internes à l'agent, ainsi que des interactions entre agents du système. Le point le plus novateur concerne la dissociation de la partie affective en trois dimensions complémentaires : les émotions, les sentiments et le tempérament de l'individu. La prise en compte d'une variabilité différente dans le temps pour chacune des ces caractéristiques psychologiques présente plusieurs avantages :

- **Le comportement de notre agent est cohérent** autant sur le plan rationnel qu'affectif. En effet, son ressenti émotionnel participe à la construction de ses interactions sociales via les sentiments qu'il va développer envers les autres individus de la simulation. Son

tempérament modère ou amplifie l'impact de ses émotions et de ses sentiments sur son comportement.

- **L'identité individuelle de chaque agent est plus marquée.** Grâce à un tempérament propre, chaque entité du système développe ses émotions et ses sentiments différemment des autres. Cela permet au fil de la simulation d'observer des comportements différenciés qui n'auraient pas pu être rendus avec la seule prise en compte des émotions.
- **On trouve une plus grande richesse de comportements observables.** Chacune des trois dimensions peut influencer à sa façon, via les règles de la base de règles, sur le processus cognitif de l'agent et sur ces interactions avec son environnement. Ainsi, on peut attribuer à chacun des états affectifs composés par ces trois dimensions des comportements différents.

Nous proposons également une organisation des règles régissant le comportement de l'agent au sein d'une base catégorisée, où chaque type de règle traduit un impact précis sur le processus cognitif. Chacune des 10 catégories est caractérisée par les éléments qui entrent en prémisses de la règle et ceux présents en conclusion. Cette catégorisation apporte plusieurs avantages :

- Une extension du nombre de règles définies aisée.
- Une abstraction du formalisme de stockage des données qui permet, à l'adaptation du module de lecture des règles près, de passer d'un langage d'expression à un autre.

Nous proposons un modèle d'interactions contextuel au système étudié. Nous distinguons les interactions avec l'environnement physique de celles avec les autres agents du système. Ceci permet une meilleure prise en compte de l'impact des dimensions comportementales qui n'ont par le même effet selon qu'on interagit avec un autre individu ou avec la chaîne de production.

Nous développons un modèle de processus de contrôle d'un agent BDI à obligation ouverte dissociant l'aspect communication de l'aspect action de l'agent. Ce processus présente l'originalité d'intégrer directement la dimension affective de l'agent dans nombre de ses étapes de décision. Ainsi, le choix d'un objectif aussi bien de communication que d'action dépend autant des connaissances rationnelles de l'agent acquises sur son environnement que de son état psychologique interne.

Une méthode de fouille adaptée aux données de simulation

Nous proposons une adaptation de la méthode des projections aléatoires pour la recherche de motifs séquentiels. La particularité de nos travaux consiste à nous intéresser aux motifs dits « non consécutifs ». Ainsi, nous ne recherchons pas comme à l'accoutumé des suites de vecteurs consécutifs (X, Y, Z, T) mais des suites moins strictes au milieu desquelles nous acceptons des vecteurs quelconques $(X, *, Z, T)$. Cette approche nous permet de détecter qu'un événement à un instant t est souvent suivi à l'instant $t + \delta$ d'un autre événement quelles que soient les étapes intermédiaires.

Nous adaptons également le prétraitement des données aux spécificités des données de simulation (non bruitées et sans données manquantes). Cela nous permet d'une part de nous affranchir de la gestion fastidieuse des données manquantes et donc de tirer pleinement partie de la simulation. D'autre part, nous gagnons en rapidité sur l'exécution de cette étape.

Perspectives

Les principales perspectives de recherches sur ces travaux concernent à la fois l'amélioration de la simulation afin d'avoir des données de plus en plus fiables et un modèle de plus en plus générique, et la poursuite des recherches de motifs fréquents dans les données de simulation.

Il serait intéressant d'étudier les possibilités offertes par la programmation orientée Aspects pour modifier dynamiquement le comportement a priori rationnel d'un agent purement cognitif en lui adjoignant des caractéristiques et des règles émotionnelles. En effet, la programmation par Aspects permet de rajouter des comportements transverses dans une application sans en modifier directement le code. Nous pourrions ainsi à partir d'un agent cognitif de référence, générer de nouveaux agents en définissant leur comportement émotionnel.

Afin que la simulation soit adaptable à différents cas d'études, une possibilité serait de rendre générique le modèle d'environnement, sa perception par les agents et les actions pouvant y être effectuées. Cette étape est indispensable à la diffusion de la simulation et à une meilleure validation par la multiplication de ses applications.

Nous n'avons pas dans nos travaux intégré de processus de planification complexe des actions comme on peut le trouver dans certains agents. Une piste intéressante pourrait donc consister à étendre notre modèle pour intégrer cette planification en y incorporant l'influence des émotions sur le processus de planification.

Au niveau de la fouille des données de simulation, une possibilité serait d'étendre le périmètre de fouille à des données concernant plusieurs agents, voir plusieurs itérations de la simulation. L'adaptation de la méthode de fouille pour y parvenir présenterait l'intérêt de résultats plus riches et au support plus large puisqu'obtenu sur des données plus nombreuses.

Enfin, il paraît évident que pour l'appropriation des résultats de fouille par un utilisateur non expert des méthodes d'ECD, la mise à disposition d'une interface conviviale permettant d'explorer ces résultats serait indispensable.

Bibliographie

Proceedings of Conference on Temporal Aspects in Information Systems, Sophia-Antipolis, 1987.

Proceedings of TIME'95 International Workshop on Temporal Representation and Reasoning, 1995.

TSQL2 Working documents, 2004. <ftp://ftp.cs.arizona.edu/tsql>.

M. Adiba, B. Q. Ngoc, and C. Collet. Aspects temporels, historiques et dynamiques dans les bases de données. *T.S.I., Numéro spécial Base de données*, 6(5) :457–478, 1987.

R. Agrawal and R. Srikant. Fast algorithms for mining association rules. In J.B. Bocca, M. Jarke, and C. Zaniolo, editors, *20th International Conference on Very Large Data Bases, VLDB'94*, pages 487–499. Morgan Kaufmann, 1994.

R. Agrawal and R. Srikant. Mining sequential patterns. In *ICDE*, pages 3–14, 1995.

R. Agrawal, K. I. Lin, H. S. Sawhney, and K. Shim. Fast similarity search in the presence of noise, scaling and translation in time series databases. In *Proceedings of the 21st International Conference on Very Large Data Bases (VLDB'95)*, pages 490–501, 1995a.

R. Agrawal, G. Psaila, E. L. Wimmers, and M. Zait. Querying shapes of histories. In *Proceedings of the 21st International Conference on Very Large Databases*, Zurich, Switzerland, 1995b.

Rakesh Agrawal and Tomasz Imielinski. Mining association rules between sets of items in large databases. pages 207–216, 1993.

Philip E. Agre and David Chapman. Pengi : An implementation of a theory of activity. In *AAAI*, pages 268–272, 1987.

J. F. Allen. Maintaining knowledge about temporal intervals. *Communication of the ACM*, 26 (11), 1983.

J. Alon, S. Sclaroff, G. Kollios, and V. Pavlovic. Discovering clusters in motion time series data. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, number I, pages 375–381, Madison, Wisconsin, 2003.

- C. M. Antunes and A. L. Oliveira. Temporal Data Mining : an Overview. In *Proceedings of the Workshop on Temporal Data Mining at the 7th ACM SIGKDD'01 International Conference on Knowledge Discovery and Data Mining*, 2001.
- G. Ariav. A Temporally Oriented Data Model. *ACM Transactions on Database Systems*, 11 (4) :499–527, 1986.
- M. B. Arnold. An excitatory theory of emotion. In M. L. Reymert, editor, *Feelings and emotions : The Mooseheart Symposium in cooperation with the university of Chicago*, pages 11–33, New York, 1950. McGraw-Hill.
- M. J. Atallah, R. Gwadera, and W. Szpankowski. Detection of significant sets of episodes in event sequences. In *Proceedings of the 4th IEEE International Conference on Data Mining (ICDM'04)*, pages 3–10, Brighton, UK, 2004.
- J. L. Austin. *How to do things with words*. Oxford University Press, 1962.
- A. D. Baddeley and G. Hitch. Working memory. *The psychology of learning and motivation : Advances in research and theory*, 8 :47–89, 1974.
- R. A. Baeza-Yates. Searching subsequences. *Theory of Computer Science*, 78 :363–376, 1991.
- P. Baldi, Y. Chauvin, T. Hunkapiller, and M. McClure. Hidden markov models of biological primary sequence information. In *Proceedings of the Natural Academy of Science*, pages 1059–1063, 1994.
- B. Bauer, J. P. Müller, and J. J. Odell. AgentUML : a Formalism for Specifying Multiagent Interactions. In Paolo Ciancarini and Michael Wooldridge eds., editor, *Proceedings of the Agent-Oriented Software Engineering Workshop*, pages 91–103, 2001.
- Fabio Luigi Bellifemine, Giovanni Caire, and Dominic Greenwood. *Developing Multi-Agent Systems with JADE*. John Wiley and Sons, 2007.
- C. Benavent. Leçon 5 : Séries Chronologiques, notes de cours. IAE de Lille, 1999. [http ://christophe.benavent.free.fr/cours/dea/DEA.htm](http://christophe.benavent.free.fr/cours/dea/DEA.htm).
- Peter L. Berger and Thomas Luckmann. *The Social Construction of Reality*. Doubleday, New York, 1966.
- M. Bernadet. Logiques non classiques, notes de cours. École Polytechnique de l'université de Nantes, 2002.
- C. Bettini, X. S. Wang, S. Jajodia, and J. L. Lin. Discovering frequent event patterns with multiple granularities in time sequences. *IEEE Transactions on Knowledge and Data Engineering*, 10 :222–237, 1998.
- L. Bisognin and S. Pesty. Agents, langage et émotions : un prototype d'agent émotionnel. In *Journées ATALA, "AGENTAL" Agents et Langues*, 2004.

-
- J. Blanchard. *Un système de visualisation pour l'extraction, l'évaluation, et l'exploration interactives des règles d'association*. PhD thesis, École Polytechnique de l'Université de Nantes, 2005.
- Alan H. Bond and Les. Gasser, editors. *Readings in Distributed Artificial Intelligence*. Morgan Kaufmann Publishers, Inc., San Mateo, California, 1988.
- G. E. P. Box, G. M. Jenkins, and G. C. Reinsel. *Time series analysis : Forecasting and control*. Pearson Education Inc., Singapore, 1994.
- M. E. Bratman. *Intentions, plans and pratical reason*. Harvard University Press, 1987.
- M. E. Bratman, D. J. Israel, and M. E. Pollack. Plans and resource-bounded practical reasoning. *Computational Intelligence*, 4(4) :349–355, 1988.
- R. A. Brooks. A robust layered control system for a mobile robot. *IEEE Journal of Robotics and Automation*, 2(1) :14–23, 1986.
- R. A. Brooks. Intelligence without reason. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 569–595, 1991.
- J. Buhler and M. Tompa. Finding motifs using random projections. *Journal of Computational Biology*, 9(2) :225–242, 2002.
- I. Cadez, D. Heckerman, C. Meek, P. Smyth, and S. White. Model-based clustering and visualisation of navigation patterns on a web site. Technical Report CA 92717-3425, Department of Information and Computer Science, University of California, Irvine, CA, 2000.
- A. Camurri and A. Coglio. An Architecture for Emotional Agents. *IEEE Multimedia*, 5(4) : 24–33, 1998.
- J. F. Canavaggio. *TEMPOS : un modèle d'historiques pour un SGBD temporel*. PhD thesis, Université Joseph Fourier de Grenoble LSR-STORM, 1997. Enter text here.
- W. B. Cannon. The james-lange theory of emotion : A critical examination and an alternative theory. *American Journal of Psychology*, 39 :100–124, 1927.
- G. Casas-Garriga. Discovering unbounded episodes in sequential data. In *Proceedings of the 7th European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD'03)*, pages 83–94, Cavtat-Dubrovnik, Croatia, 2003.
- C. Chatfield. *The analysis of time series (5th edition)*. Chapman and Hall, New York, NY, 1996.
- B. Chiu and S. Keogh, M. and Lonardi. Probabilistic discovery of time series motifs. In *Proc. of the 9th ACM International Conference on Knowledge Discovery and Data Mining (KDD'03)*, pages 493–498, Washington DC., 2003.
- V. Christophe. *Les émotions, tour d'horizon des principales théories*. Presses Universitaires du Septentrion, 1998.

- J. Clifford and A. Crocker. *Temporal Databases : Theory, Design and Implementation*, chapter The Historical Relational Data Model (HRDM) Revisited, pages 6–27. 1993.
- J. Clifford and A. Rao. A Simple, Genral Structure for Temporal Domains. In *Proceedings of the Conference on Temporal Aspect in Information Systems*, pages 23–30, 1987.
- G. Clore and K. Gasper. *Emotions and Beliefs : How Feelings Influence Thoughts*, chapter Feeling is believing : Some affective influences on belief, pages 1–9. Cambridge University Press, Cambridge, 2000.
- P. R. Cohen and H. J. Lesveque. Intention is choice with commitment. *Artificial Intelligence*, 42 :213–261, 1990.
- P. R. Cohen and C. R. Perrault. Elements of a plan based theory of speech acts. *Cognitive Science*, 3 :177–172, 1979.
- F. Corpet. Multiple sequence alignment with hierarchical clustering. *Nucleic Acids Research*, 16 :10881–10890, 1988.
- D. Cowan. An alternative to the dichotomous interpretation of jung’s psychological functions : Developing more sensitive measurement technology. In *Journal of Personality Assessment*, volume 53, pages 459–471, 1989.
- A. Damasio. *Descartes’ error : emotion, reason and the human brain*. Putnam Books, 1994.
- R. Dantzer. *Les émotions*. Presses Universitaires de France, 2003.
- T. Darrell and A. Pentland. Space-time gestures. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’93)*, pages 335–340, 1993.
- Charles Darwin. *The Expression of Emotions in Man and Animals*. 1872. Originellement publié en 1872, cet ouvrage a été plusieurs fois réédité.
- G. Das, D. Gunopulos, and H. Mannila. Finding similar time series. *Principles of Data Mining and Knowledge Discovery*, 19 :88–100, 1997.
- S. Daviet, H. Desmier, H. Briand, F. Guillet, and V. Philippe. A system of emotional agents for decision support. In Andrej Skowron, Jean-Paul Barthes, Lakhmi Jain, Ron Sun, Pierre Morizet-Mahoudeaux, Jiming Liu, and Ning Zhong, editors, *Proceedings of the 2005 IEEE/WIC/ACM International Conference on Intelligent Agent Technology (IAT’05)*, pages 711–717. IEEE Computer Society Press, 2005a. ISBN 0-7695-2416-8.
- S. Daviet, H. Desmier, and G. Rozec. Un système d’agents émotionnels pour l’aide à la décision. In *Actes du 6ème Colloque des Jeunes Chercheurs en Sciences Cognitives (CJCSC’05)*, pages 148–153, Bordeaux, France, 2–4 mai 2005b. FRESCO : Fédération Francaise des Étudiants en Sciences Cognitives. Document publié au format électronique sous Licence CC BY-NC-ND.

-
- S. Daviet, F. Guillet, A. Magda Florea, H. Briand, and V. Philippe. Modélisation des interactions entre individus avec agentuml. *Revue des Nouvelles Technologies de l'Information (RNTI-E-3)*, E : Extraction et Gestion des Connaissances(3) :Cépaduès Édition, 2005c. ISBN 2-85428-677-4.
- Stéphane Daviet. Un système d'agents émotionnels pour l'aide à la décision. In *Actes des JDOC'06*, Bordeaux, 2–4 mai 2006.
- Stéphane Daviet, Fabrice Guillet, Henri Briand, and Vincent Philippé. La simulation multi-agents pour l'aide à la décision. In *Actes des 17^{ème} journées francophones d'Ingénierie des Connaissances (IC'2006)*, Nantes, 28–30 juin 2006. Poster.
- Stéphane Daviet, Fabrice Guillet, Henri Briand, Serge Baquedano, Vincent Philippé, and Régis Gras. *Statistical Implicative Analysis*, volume 127 of *Studies in Computational Intelligence*, chapter Using the Statistical Implicative Analysis for Elaborating Behavioral Referentials, pages 299–319. Springer Berlin / Heidelberg, 2008. doi : 10.1007/978-3-540-78983-3.
- C. Delobel. *Rapport du BD3 : Bases de données, nouvelles perspectives*, chapter Les bases de données économiques, pages 31–34. 1983.
- M. V. Der Linden, X. Seron, D. L. Gall, and P. Andrès. *Neuropsychologie des lobes frontaux*. Solal, 1999.
- Christophe Dousson and Thang Vu Duong. Discovering chronicles with numerical time constraints from alarm logs for monitoring dynamic systems. In *IJCAI*, pages 620–626, 1999.
- Benjamin Drieu. L'intelligence artificielle distribuée appliquée aux jeux d'équipe situés dans un milieu dynamique : l'exemple de la RoboCup. Maîtrise d'informatique option transversale, Université Paris 8, 2001.
- Florence Duchêne. *Fusion de données multicapteurs pour un système de télésurveillance médicale de personnes à domicile*. PhD thesis, Université Joseph Fourier, Institut National Polytechnique de Grenoble, 2004.
- R.O. Duda and P. Hart. *Pattern Classification and Scene Analysis*. John Wiley & Sons, 1973.
- M. Dumas Menjivar. *TEMPOS : une plate-forme pour le développement d'applications temporelles au dessus du SGBD à objets*. PhD thesis, Université Joseph Fourier de Grenoble LSR-STORM, 2000.
- R. Durbin, S. Eddy, A. Krogh, and G. Mitchison. *Biological sequence analysis*. University Press, Cambridge, 1998.
- P. Ekman. *Disorders of the facial nerve*, chapter Facial expression and facial nerve surgery, pages 363–368. Raven Press, 1982.
- P. Ekman and W. V. Friesen. Felt, false and miserable smiles. *Felt, false and miserable smiles*, 6(4) :238–252, 1982.

- P. Ekman, W. Friesen, and P. Ellsworth. *Emotion in the human face*. Cambridge University Press, 1972.
- M. S. El-Nasr and M. Skubic. A fuzzy emotional agent for decision-making in a mobile robot. In *Proceedings of the 1998 IEEE International Conference on Fuzzy Systems*, 1998.
- O. Etzion, S. Jajodia, and S. M. Sripada. Temporal Databases : Research and Practice. *Lecture Notes in Computer Science*, 1399, 1998. Enter text here.
- W. J. Ewens and G. R. Grant. *Statistical methods in bioinformatics : An introduction*. Springer-Verlag, New York, 2001.
- M. J. Fadili, S. Ruan, D. Bloyet, and B. Mazoyer. A multistep unsupervised fuzzy clustering analysis of fmri time series. *Human Brain Mapping*, 10 :160–178, 2000.
- M.-C. Fauvet. Service Bases de Données Réparties, Chapitre 2 : Bases de Données Temporelles, notes de cours. Université Joseph Fourier, Grenoble, 2003. [http ://www-lsr.imag.fr/Les.Personnes/Marie-Christine.Fauvet/CoursMatis.html](http://www-lsr.imag.fr/Les.Personnes/Marie-Christine.Fauvet/CoursMatis.html).
- U. M. Fayyad, G. Piatetsky-Shapiro, and P. Smyth. From data mining to knowledge discovery : an overview. *Advances in knowledge discovery and data mining*, pages 495–515, 1996.
- B. Fehr and J. Russell. Concept of emotion viewed from a prototype perspective. *Journal of Experimental Psychology*, 113 :464–486, 1984.
- J. Ferber. *Les système multi-agents, vers une intelligence collective*. Inter-Éditions, 1995.
- I. A. Ferguson. *Touring Machines : An Architecture for Dynamic Rational, Mobile Agents*. PhD thesis, University of Cambridge, 1992.
- D. Fisher. Knowledge Acquisition via Incremental Conceptual Clustering. *Machine Learning*, 2 :139–172, 1987.
- D. W. Fiske. Consistency of the factorial structures of personality ratings from different sources. *Journal of abnormal and social psychology*, 44 :329–344, 1949.
- A. M. Florea, D. Kayser, and S. Pentiu. *Technologie de l’information et de la communication et appropriation des savoirs : Agents Intelligents*. Université Polytechnique de Bucarest, 2002. URL <http://turing.cs.pub.ro/auf2/>. Cours WEB Interactif.
- Yves Fossé, Stéphane Daviet, Henri Briand, and Fabrice Guillet. Représentation d’expertise psychologique sous la forme de graphes orientés, codés en rdf. In Gilbert Ritschard and Chabane Djeraba, editors, *Extraction et gestion des connaissances (EGC’2006)*, Actes des sixièmes journées d’Extraction et Gestion des Connaissances, Revue des Nouvelles Technologies de l’Information (RNTI-E-6), pages 713–714, Lille, France, 17–20 janvier 2006. Cepaduès Éditions. 2 volumes.

-
- S. Franklin and A. Graesser. Is it an Agent, or just a Program ? : a Taxonomy for Autonomous Agents. In *Proceedings of the Third International Workshop on Agent Theories, Architectures, and Languages*, pages 21–35, 1997.
- Sigmund Freud. *Œuvres complètes (1885–1938)*. Presses Universitaires de France, 2005.
- N. H. Frijda. *The Emotions*. Maison des Sciences de l’Homme and Cambridge University Press, 1986.
- J. P. Gaillard. *Psychologie de l’homme au travail : les relations homme-machine*. Dunod, 1997.
- T. Garneau and S. Delisle. Programmation orientée-agent : évaluation comparative d’outils et environnements. In Hermès Science, editor, *Actes des 10^{ème} Journées Françaises d’Intelligence Artificielle Distribuée et de Systèmes Multi-Agents JFIADSMA’02*, pages 111–123, 2002.
- M. Garofalakis, R. Rastogi, and K. Shim. Mining sequential patterns with regular expression constraints. *IEEE Transactions on Knowledge and Data Engineering*, 14 :530–552, 2002.
- Les Gasser and Michael N. Huhns, editors. *Distributed Artificial Intelligence*, volume II. Pitman Publishing, London, 1989.
- Les Gasser, Carl Braganza, and Nava Herman. *Distributed Artificial Intelligence*, chapter MACE : A Flexible Testbed for Distributed AI Research, pages 119–152. Pitman Publishers, 1987.
- Michael R. Genesereth and Nils J. Nilsson. *Logical Foundations of Artificial Intelligence*. Morgan Kaufmann, Los Altos, CA, 1987.
- J. George and G. Jones. *Organizational Behavior*. Prentic Hall, Upper Saddler River, NJ, 3rd ed. 2004 edition, 2002.
- M. P. Georgeff and A. Lansky. Reactive Reasoning and Planning : an Experiment with a Mobile Robot. In *Proceedings of the 6th National Conference on Artificial Intelligence NCAI’87*, 1987.
- A. Ghias, J. Logan, D. Chamberlin, and B. C. Smith. Query by humming : musical information retrieval in an audio database. In *Proceedins of the ACM Multimedia 95*, San Fransisco, CA, 1995.
- Piotr J. Gmytrasiewicz and Christine L. Lisetti. Using decision theory to formalize emotions. In *Proceedings of the Second Workshop on Decision Theoretic and Game Theoretic Agents, held in conjunction with the Fourth International Conference on Multi-Agent Systems IC-MAS’2000*, pages 39–47, 2000.
- B. Gold and N. Morgan. *Speech and audio signal processing : Processing and perception of speech and music*. John Wiley & Sons, New York, 2000.

- L. R. Goldberg. Language and individual differences : The search for universals in personality lexicons. *Review of Personality and Social Psychology*, 2 :Wheeler, L., 1981.
- L. R. Goldberg. An alternative "description of personality" : The big-five factor structure. *Journal of Personality and Social Psychology*, 59 :1216–1229, 1990.
- D. Goleman. *Emotional intelligence : why it can matter more than IQ ?* Bantam, 1997.
- E. Grandjean. *Précis d'ergonomie*. Les éditions d'organisation, Paris, 1985. 3^{ème} édition.
- R. M. Gray, A. Buzo, A. H. Gray Jr., and Y. Matsuyama. Distortion measures for speech processing. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 28 :367–376, 1980.
- F. Guillet and H. Hamilton, editors. *Quality measures in Data Mining*. Springer, 2007.
- D. Gusfield. *Algorithms on strings, trees and subsequences*. University of Cambridge Press, New York, 1997.
- Richard W. Hamming. Error detecting and error correcting codes. *Bell Systems Technical Journal*, 29(2) :147–160, 1950.
- J. Han and M. Kamber. *Data mining : Concepts and techniques*. Morgan Kauffmann, San Fransisco, CA, 2001.
- D. Hand, H. Mannila, and P. Smyth. *Principles of data mining*. MIT Press, Cambridge, MA, 2001.
- S. Hanks and R.J. Firby. Issues and architecture for planning and execution. In Morgan Kaufmann, editor, *Proceedings of the DARPA Workshop on Innovative Approaches to Planning, Scheduling and Control*, pages 59–70, 1990.
- Rom Harre. *The Social Construction of Emotions*. Blackwell Pub, 1989.
- E. Haselsteiner and G. Pfurtscheller. Using time-dependent neural networks for eeg classification. *IEEE Transaction Rehabilitation Engineering*, 8 :457–463, 2000.
- T. Hastie, R. Tibshirani, and J. Friedman. *The elements of statistical learning : Data mining, inference and prediction*. Springer-Verlag, New York, 2001.
- Donald O. Hebb. *Gale Encyclopedia of Psychology (2 tomes) (1904–1985)*. 2001.
- L. D. Herman, F. Hayes-Roth, Lesser V. R., and D. R. Reddy. The hearsay-ii speech-understanding system : Integrating knowledge to resolve uncertainty. *Computing Surveys*, 12(2) :213–253, 1980.
- Carl Hewitt, Peter Bishop, and Richard Steiger. A universal modular actor formalism for artificial intelligence. In *Proceedings of IJCAI*, pages 235–245, 1973.

-
- Jochen Hipp, Ulrich Güntzer, and Gholamreza Nakhaeizadeh. Algorithm for association rule mining – a general survey and comparison. In *International Conference on Knowledge Discovery and Data Mining*, 2000.
- M. Hirao, S. Inenaga, A. Shinohara, M. Takeda, and S. Arikawa. A practical algorithm to find the best episode patterns. In *Lecture notes in computer science*, volume 2226, pages 435–440, 2001.
- P. Hong and T. S. Huang. Learning to extract multi-temporal signal patterns from a temporal signal sequence. In *Proceedings of the 15th International Conference on Pattern Recognition*, Barcelona, Spain, 2000.
- Michael N. Huhns, editor. *Distributed Artificial Intelligence*. Pitman Publishing Ltd., London, England, 1987.
- C. E. Izard. *The face of emotion*. Appleton, 1971.
- C. E. Izard. *Human Emotions*. Plenum Press, 1977.
- C. E. Izard. Facial expressions and the regulation of emotions. *Journal of Personality and Social Psychology*, 58 :487–498, 1990.
- C. E. Izard. *Emotion and facial expressions : A perspective from Differential Emotions Theory*. Cambridge University Press, 1997.
- W. James. What is emotion ? *Mind*, 9 :188–205, 1884.
- C.-S. Jensen, M. Soo, and R. Snodgrass. Unification of temporal data models. In IEEE Press, editor, *Proceedings of the Data Engineering Conference*, 1993.
- C.-S. Jensen, J. Clifford, R. Elmasri, S. Gadia, P. Hayes, and S. Jajodia. A consensus glossary of temporal database concepts. *ACM SIGMOD Record*, 23(1), 1994.
- B. H. Juang and L. Rabiner. *Fundamentals of speech recognition*. NJ : Prentice Hall, Englewood Cliffs, 1993.
- K. Kalpakis and D. G. V. Puttagunta. Distance measures for effective clustering of arima time series. In *IEEE International Conference on Data Mining (ICDM'01)*, San Jose, CA, 2001.
- E. J. Keogh and M. J. Pazzani. Scaling up dynamic time warping for datamining applications. In *Proceedings of 6th ACM SIGKDD International Conference on Knowledge Discovery and Data mining*, pages 285–289, Boston, MA, 2000.
- E. J. Keogh, S. Chu, D. Hart, and M. Pazzani. Segmenting time series : A survey and novel approach. In *Data Mining In Time Series Databases, chapter 1*, pages 1–22. Publishing Company, 2003.

- M. Keogh and M. Pazzani. An enhanced representation of time series which allows fast and accurate classification, clustering and relevant feedback. In *Proceedings of the AAAI Workshop on Predicting the Future : AI Approaches to Time Series Analysis at the 4th Conference on Knowledge Discovery in Databases and Data Mining*, pages 239–241, 1998.
- A. Ketterlin. Clustering Sequences of Complex Objects. In *Proceedings of the 3rd conference on Knowledge Discovery in Databases and Data Mining*, pages 215–218, 1997.
- Y. Kodratoff. *La fouille dans les données par la méthode d'analyse statistique implicite*, chapter Extraction de connaissances à partir des données et des textes (ECD et ECT data et text mining). École polytechnique de l'Université de Nantes Nantes, 2001.
- K. Konolige and M. E. Pollack. A representationalist theory of intention. In Morgan Kaufmann, editor, *Proceedings of the 13th International Joint Conference on Artificial Intelligence IJCAI'93*, 1993.
- J. B. Kruskal. An overview of sequence comparison : Timewarps, string edits and macromolecules. *SIAM Review*, 21 :201–237, 1983.
- A. Kundu, Y. He, and P. Bahl. Word recognition and word hypothesis generation for handwritten script : A hidden markov model based approach. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'88)*, pages 457–462, 1988.
- C. Lange. *Om Sinsbevaegelser : Et psyko-fysiologiske Studie*. Rasmussen, Copenhagen, 1885.
- M. H. Law and J. T. Kwok. Rival penalized competitive learning for model-based sequence clustering. In *IEEE International Conference on Pattern Recognition (ICPR'00)*, Barcelona, Spain, 2000.
- S. Laxman and P. S. Sastry. A survey of temporal data mining. *SADHANA : Academy Proceedings in Engineering Sciences, Special Issue on Statistical Techniques in Electrical and Computer Engineering*, 31(2) :173–198, 2006.
- S. Laxman, P. S. Sastry, and K. P. Unnikrishnan. *Temporal Data Mining Workshop Notes, SIGKDD*, chapter Generalized frequent episodes in event sequences. Edmonton, Alberta, Canada, 2002.
- S. Laxman, P. S. Sastry, and K. P. Unnikrishnan. Fast algorithms for frequent episode discovery in event sequences. In *Proceedings of the 3rd Workshop on Mining Temporal and Sequential Data*, Seattle, WA, 2004.
- S. Laxman, P. S. Sastry, and K. P. Unnikrishnan. Discovering frequent episodes and learning hidden markov models : A formal connection. *IEEE Transactions on Knowledge and Data Engineering*, 17 :1505–1517, 2005.
- Richard S. Lazarus. *The nature of emotion*, chapter Emotion as coping process, pages 249–260. Penguin Books, Baltimore MD, 1968.

-
- K. Lechilli and B. Chaib-Draa. Structures relationnelles pour les interactions entre agents. In *Actes des 4^{ème} Journées Françaises d'Intelligence Artificielle Distribuée et de Systèmes Multi-Agents JFIADSMA'96*, pages 59–71. Hermès Science, 1996.
- C.-H. Lee, M.-S. Chen, and C.-R. Lin. Progressive pattern miner : An efficient algorithm for mining general temporal association rules. *IEEE Transactions on Knowledge and Data Engineering*, 15 :1004–1017, 2003.
- D. Lenat. Beings : Knowledge as interacting experts. In *Proceedings of IJCAI*, pages 126–133, 1975.
- V. I. Levenshtein. Binary codes capable of correcting deletions, insertions and reversals. *Soviet Physics Doklady*, 10(8) :707–710, 1966.
- Yingjiu Li, Peng Ning, X. Sean, and Wang Sushil Jajodia. Discovering calendar-based temporal association rules. In *8th International Symposium on Temporal Representation and Reasoning*, pages 111–118, 2001.
- J. Lin, M. Keogh, and S. Lonardi. Finding motifs in time series. In *Proceedings of the 2nd Workshop on Temporal Data Mining, at the 8th International Conference on Knowledge Discovery and Data Mining (KDD'02)*, pages 53–68, Edmonton, Alberta, Canada, 2002a.
- M.-Y. Lin and S.-Y. Lee. Improving the efficiency of interactive sequential pattern mining by incremental pattern discovery. In *Proceedings of the IEEE 36th Annual Hawaii International Conference on System Sciences (HICSS'03)*, Big Island, Hawaii, 2003.
- W. Lin, M. A. Orgun, and G. J. Williams. An Overview of Temporal Data Mining. In *Proceedings of the 1st ADM Australian Data Mining Workshop*, pages 83–90, University of Technology, Sydney, 2002b.
- Huan Liu and Hiroshi Motoda. *Feature selection for knowledge discovery and data mining*. Kluwer Academic Publishers, 1998.
- H. Lu, J. Han, and L. Feng. Stock Price Movement and N-Dimensional Transaction Association Rules. In *Proceedings of the ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery*, number 12, pages 1–7, 1998.
- P. Lyman and H. R. Varian. How much information ? Technical report, School of Information Management and Systems at the University of California at Berkeley, 2003. URL www.sims.berkeley.edu/how-much-info-2003.
- Mac Dermott. Non monotonic logic II. *Journal of ACM*, (29) :34–57, 1982.
- Pattie Maes. The agent network architecture (ana). *SIGART Bulletin*, 2(4) :115–120, 1991.
- G. Mandler. *Cognitive science and clinical disorders*, chapter Cognition and emotion : extensions and clinical applications. Academic Press, 1992.

- H. Mannila, H. Toivonen, and A. I. Verkamo. Discovery of frequent episodes in event sequences. *Data Mining Knowledge Discovery*, 1 :259–289, 1997.
- N. Meger and C. Rigotti. Constraint-based mining of episode rules and optimal window sizes. In *Proceedings of the 8th European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD'04)*, Pisa, Italy, 2004.
- J. Méléze. *L'analyse modulaire des systèmes de gestion*, A.M.S. 1972.
- J. Mennis and J. Liu. Mining Association Rules in Spatio-Temporal Data. In *Proceedings of the 7th International Conference on Geocomputation*, 2003.
- R. T. Miller, A. G. Christoffels, C. Gopalakrishnan, J. Burke, A. A. Ptitsyn, T. R. Broveak, and W. A. Hide. A comprehensive approach to clustering of expressed human gene sequence : The sequence tag alignment and consensus knowledge base. *Genome Research*, 9 :1143–1155, 1999.
- Hans Moravec. *Mind Children - The Future of Robot and Human Intelligence*. Harvard University Press, 1988.
- I. Myers. The myers-briggs type indicator. *Educational Testing Service*, 1962.
- Isabel Briggs Myers and Peter B. Myers. *Gifts Differing : Understanding Personality Type*. Davies-Black Publishing, Mountain View, CA, 1995.
- R. Nag, K. H. Wong, and F. Fallside. Script recognition using hidden markov models. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP'86)*, pages 2071–2074, 1986.
- Osato Naoki, Itoh Masayoshi, Konno Hideaki, Kondo Shinji, Shibata Kazuhiro, Carninci Piero, Shiraki Toshiyuki, Shinagawa Akira, Arakawa Takahiro, Kikuchi Shoshi, Sato Kouji, Kawai Jun, and Hayashizaki Yoshihide. A computer-based method of selecting clones for a full-length cdna project : Simultaneous collection of negligibly redundant and variant cdnas. *Genome Research*, 12 :1127–1134, 2002.
- A. Newell. Some problems of the basic organization in problem-solving programs. In G. T. Jacoby M. C. Yovits and editors G. D. Goldstein, editors, *Proceedings of the Second Conference on Self-Organizing Systems*, pages 393–423. Spartan Books, 1962.
- R. T. Ng, L. V. S. Lakshmanan, J. Han, and A. Pang. Exploratory mining and pruning optimizations of constrained associations rules. In *Proceedings of the 1998 ACM SIGMOD International Conference on Management of Data*, pages 13–24, Seattle, Washington, 1998.
- Stéphanie Noci. Le test Sosie : fiabiliser l'évaluation des personnes : Les outils d'évaluation des personnes. *Actualité de la formation permanente*, 186 :17–20, 2003.
- D. A. Norman and T. Shallice. *Consciousness and self-regulation*, volume 4, chapter Attention to action : Willed and automatic control of behavior, pages 1–18. New York : Plenum, 1980.

-
- OMG. Unified Model Language Specification, version 2.1.2, 2007. <http://www.omg.org>.
- A. Ortony, G. Clore, and A. Collins. *The cognitive structure of emotions*. Cambridge University Press, 1988.
- D. O'Shaughnessy. *Speech communications : Human and machine*. IEEE Press, Piscataway, NJ, 2003.
- N. Pasquier, Y. Bastide, R. Taouil, and L. Lakhal. Discovering frequent closed itemsets for association rules. *Lecture Notes in Computer Science*, 1540 :398–416, 1999.
- PerformanSe. Classeur de formation performanse. Technical report, 2003.
- V. Philippé, S. Baquedano, R. Gras, P. Peter, J. Juhel, P. Vrignaud, and Y. Forner. étude de validation : Performanse echo, performanse oriente. Technical report, Study realized with the collaboration of PerformanSe, Laboratoire COD de l'École Polytechnique de l'Université de Nantes, Laboratoire de Psychologie Différentielle de l'Université de Rennes 2, 2004.
- Jean Piaget. *Introduction à l'épistémologie génétique : Tomes I – La pensée mathématiques, II – La pensée physique & III – La pensée biologique, la pensée psychologique et la pensée sociologique*. 1950.
- R. Plutchik and H. Kellerman. *Emotion : Theory, research and experience. Vol. 1 : Theories of emotion*. Academic Press, 1980.
- M.-C. Quidoz. Modèles et systèmes de bases de données temporelles et actives en économie. Mémoire d'Ingénieur CNAM, 1997. Université Joseph Fourier de Grenoble, LSR-STORM.
- R Development Core Team. *R : A language and environment for statistical computing*. R Foundation for Statistical Computing, Vienna, Austria, 2005. URL <http://www.R-project.org>. ISBN 3-900051-00-3.
- Patrick Rabbitt. *Methodology of Frontal and Executive Function*. Psychology Press, 1997.
- A. S. Rao and M. P. Georgeff. Modeling Rational Agents within a BDI-Architecture. In *Proceedings of the Second International Conference on Principle of Knowledge Representation and Reasoning KR'91*. Morgan Kaufmann, 1991.
- A. S. Rao and M. P. Georgeff. BDI Agents : from theory to practice. In *Proceedings of the First International Conference on Multi-Agent Systems*, 1995.
- J. F. Roddick. Temporal, Spatial and Spatio-Temporal Data-Mining. *Lecture Notes in Computer Science*, 2007, 2000. Revised Paper of the First International Workshop TSDM'00.
- J. F. Roddick, K. Hornsby, and M. Spiliopoulou. YABTSSTDMR, Yet Another Bibliography of Temporal, Spatial and Spatio-Temporal Data-Mining Research. In *Proceedings of the Workshop on Temporal Data Mining at the 7th SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2001.

- G. Rozec. Les processus attentionnels, les fonctions exécutives et les troubles du comportement engagés dans la réinsertion professionnelle et sociale des personnes traumatisées crâniennes. Technical report, 2003.
- S. J. Russell and P. Norvig. *Artificial Intelligence : A Modern Approach*. Prentice Hall, 1995.
- A. Sauvy. Que sais-je ? In *La population*. PUF, 1979.
- S. Schachter and J. Singer. Cognitive, social, and physiological determinants of emotional state. *Psychological Review*, 69 :379–399, 1962.
- K. R. Scherer. *The evaluation of speech in psychiatry and medicine*, chapter Speech and Emotional states. 1981.
- T. Schreiber and A. Schmitz. Classification of time series data with nonlinear similarity measures. *Physical Review Letter*, 79 :1475–1478, 1997.
- J. R. Searle. *Speech Acts : an Essay in the Philosophy of Language*. Cambridge University Press, 1969.
- J. R. Searle. Minds, brains and programs. *The Behavioral and Brain Sciences*, 3 :417–424, 1980.
- P. Sebastiani, M. Ramoni, P. Cohen, J. Warwick, and J. Davis. Discovering dynamics using bayesian clustering. *Lecture notes in computer science ; Advances in Artificial Intelligence in Intelligent Data Analysis : 3rd International Symposium (IDA'99)*, 1642, 1999.
- T. Shintani and M. Kitsuregawa. Mining algorithms for sequential patterns in parallel : Hash based approach. In *Proceedings of 2nd Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 283–294, 1998.
- Y. Shoham. Reified temporal logics : Semantical and ontological consideration. *Advances in Artificial Intelligence*, II :183–190, 1987. Elsevier Science Publishers.
- Y. Shoham. Agent-oriented programming. *Artificial Intelligence*, 60(1) :51–92, 1993. Elsevier Science Publishers Ltd.
- H. A. Simon. Motivational and Emotional Controls of Cognition. *Psychological Review*, 74 : 29–38, 1967.
- H. A. Simon. *The Sciences of the Artificial*. MIT Press : Cambridge, 1981. 2nd edition.
- P. Sinclair and S. Barrow. Identifying Personality Traits predictive of Performance. *The BPS's journal on Occupational Testing – Selection and Development Review*, 8(5), 1992.
- A. Sloman. Varieties of affect and the CogAff architecture schema. In *Proceedings of AISB'01 Symposium on Adaptive Agent and Multi-Agent Systems*, 2001.
- Craig A. Smith and Richard S. Lazarus. *Handbook of Personality : Theory and Research.*, chapter Chapter 23 : Emotion and Adaptation, pages 609–637. Guilford, New York, 1990.

-
- P. Smyth. Clustering Sequence with Hidden Markov Models. *Advances in Neural Information Processing Systems*, 9 :648–654, 1997. The MIT Press.
- P. Smyth. Probabilistic Model-Based Clustering of Multivariate and Sequential Data. In *Proceedings of the Seventh International Workshop on Artificial Intelligence and Statistics*, pages 299–304, 1999.
- R. T. Snodgrass. Overview of the special session on temporal database infrastructure. *ACM SIGMOD RECORD*, 23(1), 1994.
- R. T. Snodgrass. *The TSQL2 Temporal Query Language*. Kluwer Academic Publishers, 1995.
- R. Srikanth and R. Agrawal. Mining sequential patterns : Generalizations and performance improvements. In *Proceedings 5th International Conference on Extending Database Technology (EDBT)*, Avignon, France, 1996.
- T. E. Starner and A. Pentland. Visual recognition of american sign language. In *International Workshop on Face and Gesture Recognition*, 1995.
- B. S. Stein and J. D. Bransford. Constraints on effective elaboration : effects of precision and subject generation. *Journal of Verbal Learning and Verbal Behavior*, 18 :769–777, 1979.
- A. U. Tansel, J. Clifford, S. Gadia, S. Jajodia, A. Segev, and R. T. Snodgrass. *Temporal Databases : Theory, Design and Implementation*. Benjamin Cummings, 1993.
- C. C. Tappert, C. Y. Suen, and T. Wakahara. The state of the art in on-line handwriting recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12 :787–808, 1990.
- Wei-Guang Teng, Ming-Syan Chen, and Philip S. Yu. A regression-based temporal pattern mining scheme for data streams. In *29th International Conference on Very Large Data Bases (VLDB'03)*, Berlin, Germany, 2003.
- The Foundation for Intelligent Physical Agents. Fipa acl message structure specification. Specification, 2002.
- P. Tino, C. Schittenkopf, and G. Dorffner. Temporal pattern recognition in noisy non-stationary time series based on quantization into symbolic streams : Lessons learned from financial volatility trading. Technical report, Institut für Informationsverarbeitung und-wirtschaft (Wien), 2000.
- S. P. Tipper. Less attentional selectivity as a result of declining inhibition in older adults. *Psychonomic Society Bulletin*, 29 :571–590, 1991.
- S. S. Tomkins. *Affect, imagery, consciousness. Vol. 1 : The positive affects*, volume 1. Springer Verlag, New York, 1962.

- S. S. Tomkins. *Emotion : Theory, research and experience. Vol. 1 : Theories of emotion*, volume 1, chapter Affect as amplification : Some modifications in theory. Academic Press, New York, 1980.
- S. S. Tomkins. *The emergence of personality*, chapter Script theory, pages 147–216. Springer, New York, 1986.
- Z. Tronicek. Episode matching. In *Proceedings of the 12th Annual Symposium on Combinatorial Pattern Matching (CPM'01)*, volume 2089, pages 143–146, Jerusalem, Israel, 2001.
- E. Tulving. *Organization of Memory*, chapter Episodic and semantic memory, pages 381–403. New York : Academic Press, 1972.
- E. C. Tupes and R. E. Christal. Recurrent personality factors based on trait ratings. USAF ASD Tech. Rep. No. 61-97, TX : U. S. Air Force., Lackland Airforce Base, 1961.
- A. M. Turing. Computing machinery and intelligence. *Mind*, 59 :433–460, 1950.
- Gary VandenBos. *APA Dictionary of Psychology*. American Psychological Association, Washington DC, 2006.
- J. Velásquez. Modeling Emotions and Other Motivations in Synthetic Agents. In *Proceedings of the AAAI Conference*, pages 10–15, 1997.
- Keshri Verma and O. P. Vyas. Efficient calendar based temporal association rule. *SIGMOD Rec.*, 34(3) :63–70, 2005. ISSN 0163-5808. doi : <http://doi.acm.org/10.1145/1084805.1084818>.
- L. von Bertalanffy. *Théorie générale des systèmes*. 1973.
- J. Wang and J. Han. Bide : Efficient mining of frequent closed sequences. In *Proceedings of the 20th International Conference on Data Engineering*, Boston, MA, 2004.
- John Broadus Watson. Psychology as the behaviorist views it. *Psychological Review*, 20 : 158–177, 1913.
- J. S. Wiggins, editor. *The Five-Factor Model of Personality : Theoretical Perspectives*. Guilford, New York, 1996.
- I. H. Witten and E. Frank. *Data mining : Practical machine learning tools and techniques with JAVA implementations*. Morgan Kaufmann, San Fransisco, CA, 2000.
- M. Wooldridge. *An introduction to multi-agent systems*. John Wiley and Sons Ltd. Ed., 2001.
- M. Wooldridge and N. Jennings. Intelligent agents : theory and practice. *Knowledge Engineering Review*, 10(2), 1995a. Cambridge University Press.
- M. Wooldridge, M. d’Inverno, D. Kinny, and M. Luck. A Formal Specification of dMARS. In *Proceedings of the 4th International Workshop on Agent Theories, Architectures and Languages ATAL'97*, 1997.

-
- Michael Wooldridge and Nicholas R. Jennings, editors. *Intelligent Agents, ECAI-94 Workshop on Agent Theories, Architectures and Languages*, volume 890 of *Lecture Notes in Computer Science*, Amsterdam, Netherlands, 1995b. Springer. ISBN 3-540-58855-8.
- C. Wu, M. Berry, S. Shivakumar, and J. McLarty. Neural networks for full-scale protein sequence classification : Sequence encoding with singular value decomposition. *Machine Learning, Special issue on applications in molecular biology*, 21(1–2) :177–193, 1995.
- S. Wu and U. Manber. Fast text searching allowing errors. *Communication of the ACM*, 35(10) :83–91, 1992.
- Y.-L. Wu, D. Agrawal, and A. E. Abbadi. Acomparison of dft and dwt based similarity search in time series databases. In *Proceedings of the 9th International Conference on Information and Knowledge Management*, pages 488–495, McLean, VA, 2000.
- Y. Xiong and D. Y. Yeung. Mixtures of arma models for model-based time series clustering. In *IEEE International Conference on Data Mining*, pages 717–720, Maebashi City, Japan, 2002.
- J. Yamato, J. Ohya, and K. Ishii. Recognizing human action in time-sequential images using hidden markov model. In *IEEE Comput. Soc. Conf. on Computer Vision and Pattern Recognition (CVPR'92)*, pages 379–385, 1992.
- X. Yan, J. Han, and R. Afshar. Clospan : Mining closed sequential patterns in large datasets. In *Proceedings of the International SIAM Conference on Data Mining (SDM'03)*, San Francisco, CA, 2003.
- G. U. Yule. On a method of investigating periodicity in distributed series, with special reference to wolfer's sunspot numbers. *Philosophical Transaction of the Royal Society of London*, A226 :267–298, 1927.
- M. J. Zaki. Efficient enumeration of frequent sequences. In *Proceedings of the ACM 7th International Conference Information and Knowledge Management (CIKM)*, 1998.

Annexes

Modèle OCC : déclenchement des émotions

Les émotions de bien-être

TAB. 1 – Les émotions de bien-être.

	Évaluation de l'évènement qui arrive	
	désirable	indésirable
Déclenchement de l'émotion	<i>joie</i>	<i>tristesse</i>
Variables qui influent sur l'intensité	Degré de désirabilité de l'évènement	Degré de non-désirabilité de l'évènement

Les émotions liées aux ressentis pour les autres

TAB. 2 – Les émotions liées aux ressentis pour les autres.

Réaction personnelle		Désir supposé de l'évènement qui arrive pour un autre	
		désirable	indésirable
plaisir	Déclenchement de l'émotion	<i>content pour</i>	<i>joie malsaine</i>
	Variables qui influent sur l'intensité	Degré d'envie pour nous que l'évènement arrive à l'autre Degré supposé du désir de l'évènement par l'autre Degré de mérite de l'évènement pour l'autre Degré d'amitié avec l'autre	Degré d'envie pour nous que l'évènement arrive à l'autre Degré supposé du non-désir de l'évènement par l'autre Degré de mérite de l'évènement pour l'autre Degré d'inimitié avec l'autre
déplaisir	Déclenchement de l'émotion	<i>jalousie</i>	<i>pitié</i>
	Variables qui influent sur l'intensité	Degré de non-envie pour nous que l'évènement arrive à l'autre Degré supposé du désir de l'évènement par l'autre Degré de non-mérite de l'évènement pour l'autre Degré d'inimitié avec l'autre	Degré de non-envie pour nous que l'évènement arrive à l'autre Degré supposé de non-désir de l'évènement par l'autre Degré de non-mérite de l'évènement pour l'autre Degré d'amitié avec l'autre

Les émotions liées à un évènement éventuel

TAB. 3 – Les émotions liées à un évènement éventuel.

Statut de l'évènement		Évaluation d'un évènement éventuel	
		désirable	indésirable
non confirmé	Déclenchement de l'émotion	<i>espoir</i>	<i>peur</i>
	Variables qui influent sur l'intensité	Degré de désir pour l'évènement Probabilité de l'évènement	Degré de non-désir pour l'évènement Probabilité de l'évènement
confirmé	Déclenchement de l'émotion	<i>satisfaction</i>	<i>horreur</i>
	Variables qui influent sur l'intensité	Intensité de l'émotion espoir liée au même évènement Effort fourni pour atteindre l'évènement Degré de réalisation	Intensité de l'émotion peur liée au même évènement Effort fourni pour prévenir l'évènement Degré de réalisation
infirmé	Déclenchement de l'émotion	<i>déception</i>	<i>soulagement</i>
	Variables qui influent sur l'intensité	Intensité de l'émotion espoir liée au même évènement Effort fourni pour atteindre l'évènement Degré de réalisation de l'évènement	Intensité de l'émotion peur liée au même évènement Effort fourni pour prévenir l'évènement Degré de réalisation de l'évènement

TAB. 4 – Les émotions d’attribution.

Identité de l’agent		Évaluation de l’action d’un agent	
		louable <i>fierté</i>	blâmable <i>honte</i>
soi	Déclenchement de l’émotion	Degré de louabilité de l’évènement Force de l’unité cognitive Probabilité que l’on fasse cette action face à cet évènement	Degré de blâme de l’évènement Force de l’unité cognitive Probabilité que l’on fasse cette action face à cet évènement
	Variables qui influent sur l’intensité		
autrui	Déclenchement de l’émotion	<i>engouement</i>	<i>indignation</i>
	Variables qui influent sur l’intensité	Degré de louabilité de l’évènement Probabilité que l’agent fasse cette action face à cet évènement	Degré de blâme de l’évènement Probabilité que l’agent fasse cette action face à cet évènement

Les émotions d’attribution

Les émotions composées : bien-être et attribution

TAB. 5 – Les émotions composées : bien-être et attribution.

Émotions attribution		Émotions de bien-être	
		joie <i>gratitude</i>	détresse
admiration	Déclenchement de l’émotion	Degré de louabilité de l’évènement Probabilité que l’on fasse cette action face à cet évènement Degré d’envie de l’évènement	
	Variables qui influent sur l’intensité		
reproche	Déclenchement de l’émotion		<i>colère</i> Degré de blâme de l’évènement Probabilité que l’on fasse cette action face à cet évènement Degré de non-envie de l’évènement
	Variables qui influent sur l’intensité		
fierté	Déclenchement de l’émotion	<i>gratification</i> Degré de louabilité de l’évènement Force de l’unité cognitive Probabilité que l’on fasse cette action face à cet évènement Degré d’envie de l’évènement	
	Variables qui influent sur l’intensité		
honte	Déclenchement de l’émotion		<i>remord</i> Degré de blâme de l’évènement Force de l’unité cognitive Probabilité que l’on fasse cette action face à cet évènement Degré de non-envie de l’évènement
	Variables qui influent sur l’intensité		

Les émotions d’interaction

TAB. 6 – Les émotions d’interaction.

	Évaluation de l’objet	
	attirant <i>goût</i>	repoussant <i>dégoût</i>
Déclenchement de l’émotion	Degré d’attrance de l’objet Degré de familiarité avec l’objet	Degré de répulsion pour l’objet Degré de familiarité avec l’objet
Variables qui influent sur l’intensité		

Extrait des règles psychologiques

Émotion ressentie	Tempérament			Sentiments		
JOIE	ANX	COM		Quiétude	Amitié	Confiance
MIN	MIN	MIN	→	0 ou -	0 ou -	0 ou -
MIN	MIN	NEUTRE	→	0 ou -	-	-
MIN	MIN	MAX	→	0 ou -	--	-
MIN	NEUTRE	MIN	→	-	-	-
MIN	NEUTRE	NEUTRE	→	-	-	-
MIN	NEUTRE	MAX	→	-	-	--
MIN	MAX	MIN	→	--	0	-
MIN	MAX	NEUTRE	→	--	-	-
MIN	MAX	MAX	→	--	--	--

Émotion ressentie	Tempérament			Sentiments		
JOIE	ANX	COM		Quiétude	Amitié	Confiance
NEUTRE	MIN	MIN	→	0	0	0
NEUTRE	MIN	NEUTRE	→	0	0	0
NEUTRE	MIN	MAX	→	0	0	0
NEUTRE	NEUTRE	MIN	→	0	0	0
NEUTRE	NEUTRE	NEUTRE	→	0	0	0
NEUTRE	NEUTRE	MAX	→	0	0	0
NEUTRE	MAX	MIN	→	0	0	0
NEUTRE	MAX	NEUTRE	→	0	0	0
NEUTRE	MAX	MAX	→	0	0	0

Émotion ressentie	Tempérament			Sentiments		
JOIE	ANX	COM		Quiétude	Amitié	Confiance
MAX	MIN	MIN	→	0 ou +	++	++
MAX	MIN	NEUTRE	→	0 ou +	+	++
MAX	MIN	MAX	→	0 ou +	+	+
MAX	NEUTRE	MIN	→	+	++	++
MAX	NEUTRE	NEUTRE	→	+	+	++
MAX	NEUTRE	MAX	→	+	+	+
MAX	MAX	MIN	→	++	++	++
MAX	MAX	NEUTRE	→	++	0	++
MAX	MAX	MAX	→	++	0	+

2. Extrait des règles psychologiques

Émotion ressentie	Tempérament			Sentiments		
GOÛT	ANX	COM	→	Quiétude	Amitié	Confiance
MIN	MIN	MIN	→	0	-	0
MIN	MIN	NEUTRE	→	0	-	-
MIN	MIN	MAX	→	0	--	--
MIN	NEUTRE	MIN	→	0 ou -	-	0
MIN	NEUTRE	NEUTRE	→	0 ou -	-	-
MIN	NEUTRE	MAX	→	0 ou -	--	-
MIN	MAX	MIN	→	-	-	--
MIN	MAX	NEUTRE	→	-	-	--
MIN	MAX	MAX	→	-	-	--

Émotion ressentie	Tempérament			Sentiments		
GOÛT	ANX	COM	→	Quiétude	Amitié	Confiance
NEUTRE	MIN	MIN	→	0	0	0
NEUTRE	MIN	NEUTRE	→	0	0	0
NEUTRE	MIN	MAX	→	0	0	0
NEUTRE	NEUTRE	MIN	→	0	0	0
NEUTRE	NEUTRE	NEUTRE	→	0	0	0
NEUTRE	NEUTRE	MAX	→	0	0	0
NEUTRE	MAX	MIN	→	0	0	0
NEUTRE	MAX	NEUTRE	→	0	0	0
NEUTRE	MAX	MAX	→	0	0	0

Émotion ressentie	Tempérament			Sentiments		
GOÛT	ANX	COM	→	Quiétude	Amitié	Confiance
MAX	MIN	MIN	→	0	++	++
MAX	MIN	NEUTRE	→	0	+	++
MAX	MIN	MAX	→	0	+	+
MAX	NEUTRE	MIN	→	+	++	++
MAX	NEUTRE	NEUTRE	→	+	+	++
MAX	NEUTRE	MAX	→	+	+	+
MAX	MAX	MIN	→	++	++	++
MAX	MAX	NEUTRE	→	++	+	++
MAX	MAX	MAX	→	++	+	++

Émotion ressentie	Tempérament			Sentiments		
GRATITUDE	ANX	COM	→	Quiétude	Amitié	Confiance
MIN	MIN	MIN	→	0	-	-
MIN	MIN	NEUTRE	→	0	0	-
MIN	MIN	MAX	→	0	--	--
MIN	NEUTRE	MIN	→	-	-	-
MIN	NEUTRE	NEUTRE	→	-	0	-
MIN	NEUTRE	MAX	→	-	--	--
MIN	MAX	MIN	→	--	-	-
MIN	MAX	NEUTRE	→	--	-	--
MIN	MAX	MAX	→	--	--	--

Émotion ressentie	Tempérament			Sentiments		
GRATITUDE	ANX	COM	→	Quiétude	Amitié	Confiance
NEUTRE	MIN	MIN	→	0	0	0
NEUTRE	MIN	NEUTRE	→	0	0	0
NEUTRE	MIN	MAX	→	0	0	0
NEUTRE	NEUTRE	MIN	→	0	0	0
NEUTRE	NEUTRE	NEUTRE	→	0	0	0
NEUTRE	NEUTRE	MAX	→	0	0	0
NEUTRE	MAX	MIN	→	0	0	0
NEUTRE	MAX	NEUTRE	→	0	0	0
NEUTRE	MAX	MAX	→	0	0	0

Émotion ressentie	Tempérament			Sentiments		
GRATITUDE	ANX	COM	→	Quiétude	Amitié	Confiance
MAX	MIN	MIN	→	+	++	++
MAX	MIN	NEUTRE	→	+	+	++
MAX	MIN	MAX	→	+	+	+
MAX	NEUTRE	MIN	→	+	++	++
MAX	NEUTRE	NEUTRE	→	+	+	++
MAX	NEUTRE	MAX	→	+	+	+
MAX	MAX	MIN	→	++	++	++
MAX	MAX	NEUTRE	→	++	+	++
MAX	MAX	MAX	→	++	+	+

Modélisation AgentUML

Diagramme UML des caractéristiques personnelles

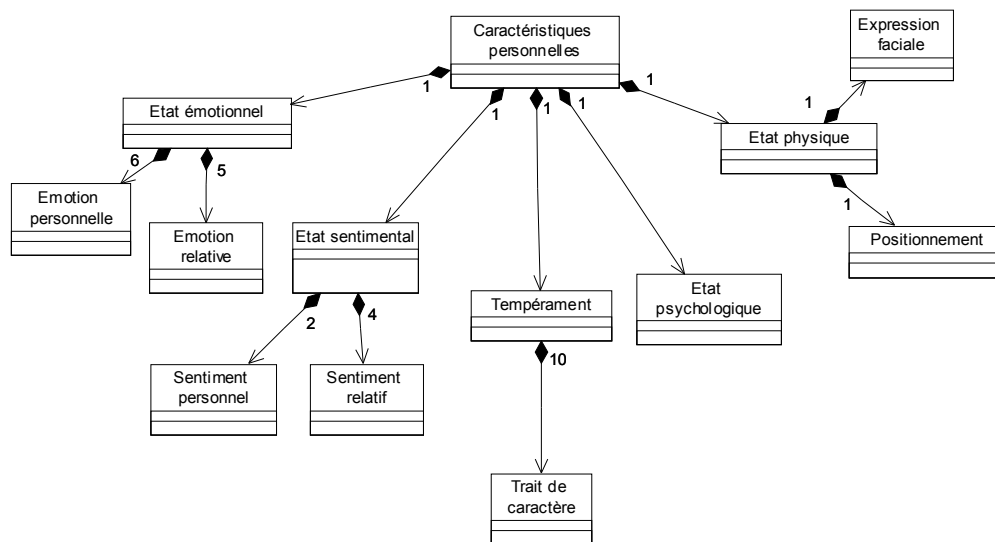


Diagramme UML de la mémoire

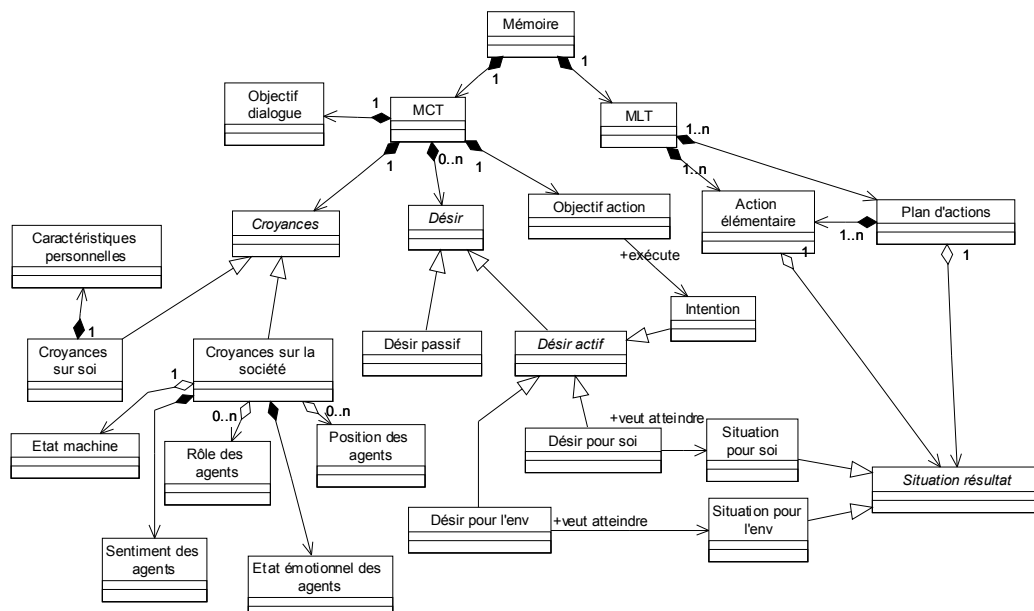


Diagramme UML du système procédural

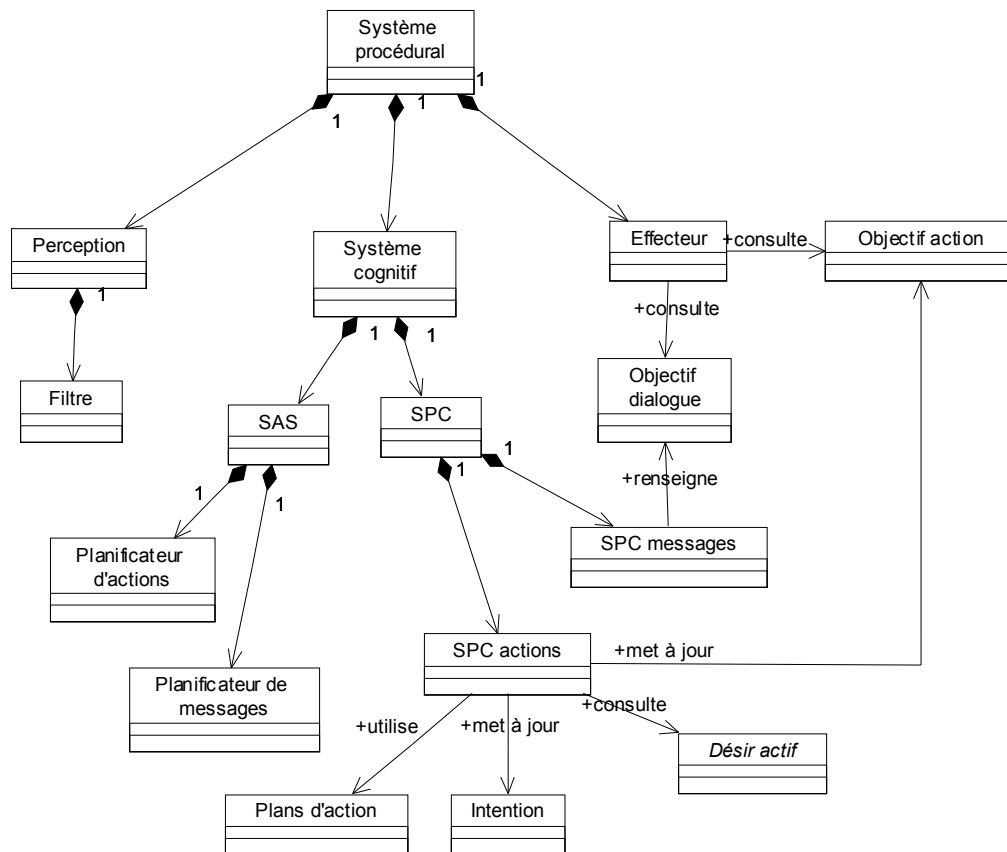


FIG. – Diagramme UML de classe du système procédural.

Diagramme de cas d'utilisation du poste 7

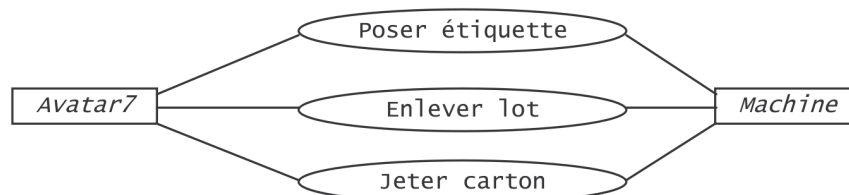


FIG. – Diagramme UML de cas d'utilisation du poste 7.

Diagrammes AgentUML de séquence du poste 7

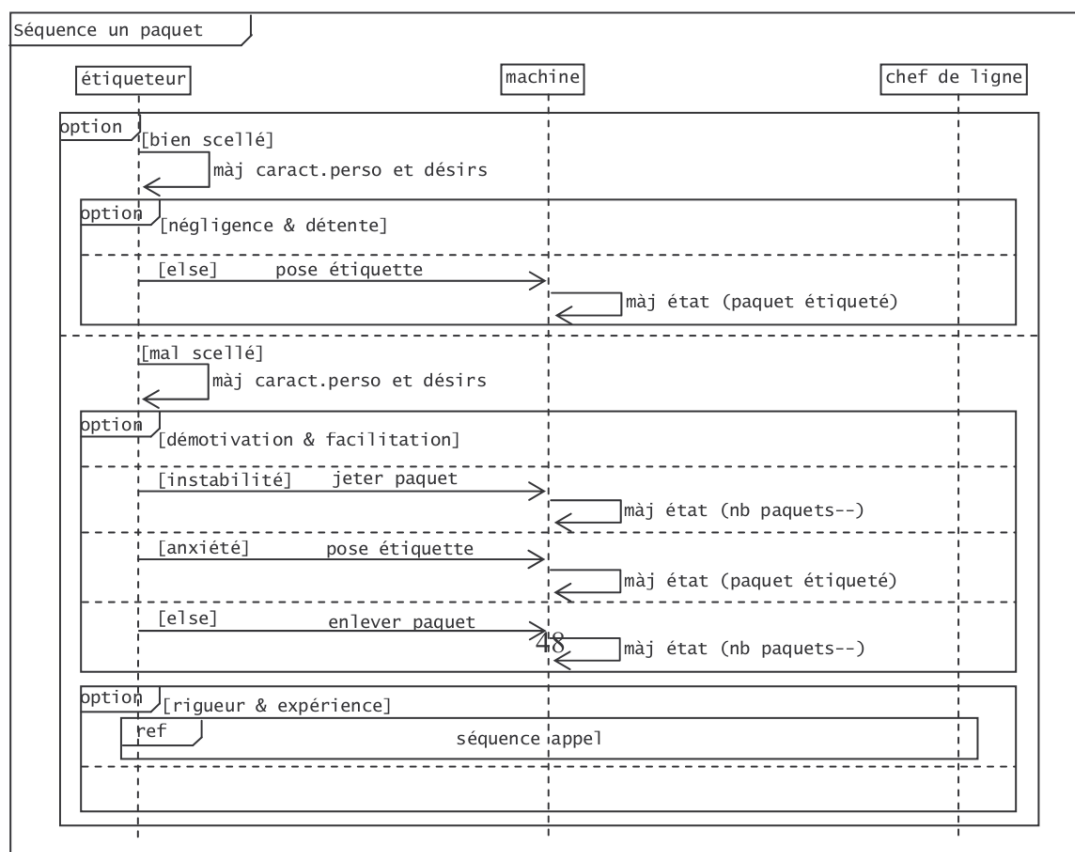


FIG. – Diagramme AgentUML de séquence du poste 7 : cas 1 paquet.

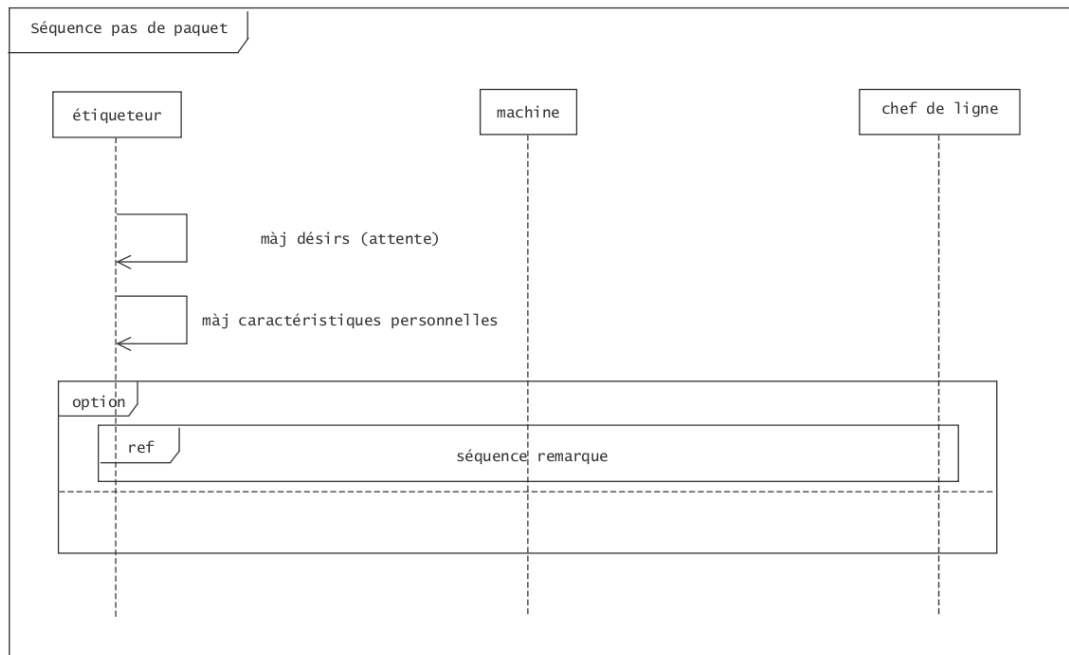


FIG. – Diagramme AgentUML de séquence du poste 7 : cas pas de paquet.

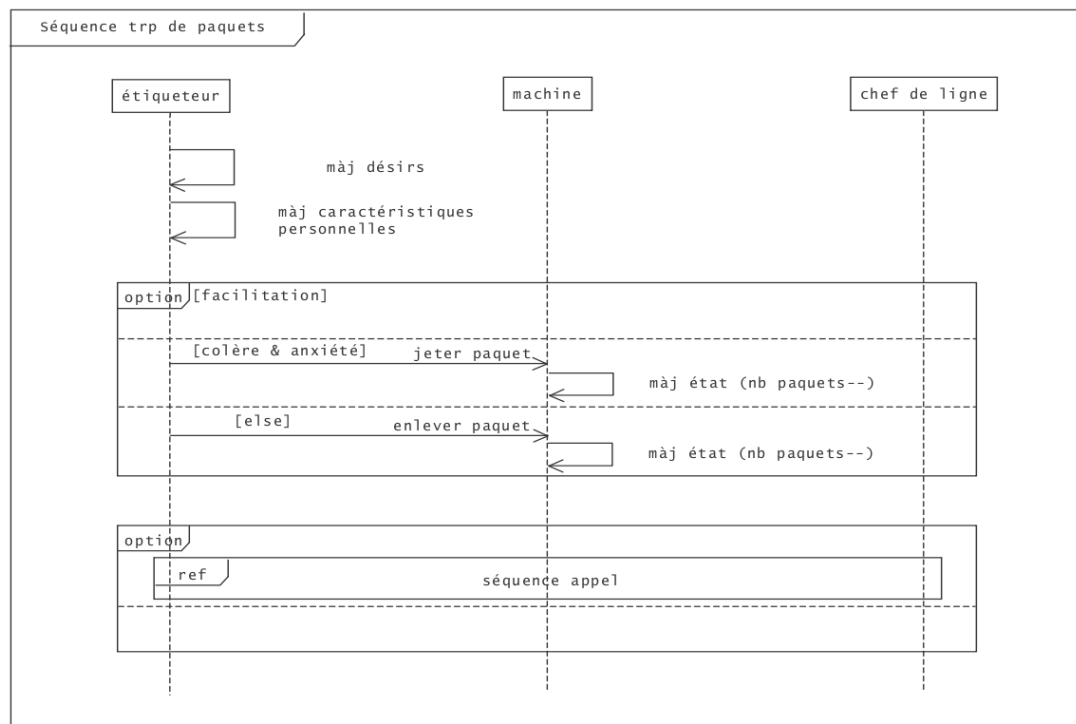


FIG. – Diagramme AgentUML de séquence du poste 7 : cas trop de paquets.

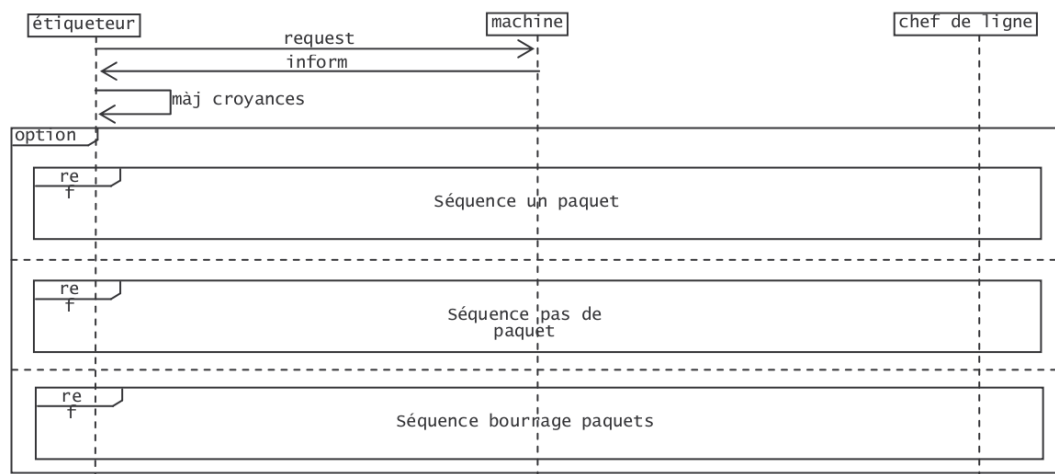


FIG. – Diagramme AgentUML de séquence du poste 7 : cas global.

Références des articles présentés en annexe

- S. Daviet, H. Desmier, H. Briand, F. Guillet, and V. Philippe. A system of emotional agents for decision support. In Andrej Skowron, Jean-Paul Barthes, Lakhmi Jain, Ron Sun, Pierre Morizet-Mahoudeaux, Jiming Liu, and Ning Zhong, editors, *Proceedings of the 2005 IEEE/WIC/ACM International Conference on Intelligent Agent Technology (IAT'05)*, pages 711–717. IEEE Computer Society Press, 2005a. ISBN 0-7695-2416-8.
- S. Daviet, H. Desmier, and G. Rozec. Un système d'agents émotionnels pour l'aide à la décision. In *Actes du 6ème Colloque des Jeunes Chercheurs en Sciences Cognitives (CJCSC'05)*, pages 148–153, Bordeaux, France, 2–4 mai 2005b. FRESCO : Fédération Française des Étudiants en Sciences Cognitives. Document publié au format électronique sous Licence CC BY-NC-ND.

A system of emotional agents for decision-support

Stéphane Daviet^{1,2}, Hélène Desmier^{1,2}, Henri Briand¹, Fabrice Guillet¹, Vincent Philippé²

¹LINA, ²PerformanSe

stephane.daviet@knowesia.fr, helene.desmier@performanse.fr

Abstract

To facilitate the study of numerous phenomena, we use computer simulation tools. However, simulation of human behavior is still a challenge for both computer and human sciences. Multi-agent systems have already proved their efficiency to simulate complex interacting systems. In this context, we have worked on the simulation of human groups to study the emergence of behavioral patterns, mixing techniques of artificial intelligence, multi-agent systems and psychological sciences. This paper presents a new model of emotional agent, the Emotion, Feeling, Temperament agent (EFT). Based on BDI architecture, our model integrates the OCC emotional model and the PerformanSe behavioral model. We also present a concrete implementation of our model: the simulation of brain-damaged people's behavior on a production line. Then, we describe the modeling of the interactions of this system with AgentUML. The final goal of our work is to produce simulation data than can be enhanced through a KDD process.

1. Introduction

Usually in decision-support computing, tools are used to analyze the existing data of a company to extract knowledge [23]. In the “Tc&Plus.Virtuel” project, our objective is to build a decision-support tool for a company whose role is to accompany brain-damaged people in reintegrating the working world. Consequently, our goal is to get knowledge about these people working in a workshop and about their psychological evolution over a working day. Observing them *in vivo* would be very long and difficult because of the influence of the observer on the behavior of brain damaged people. A computer simulation, based on a multi-agent system [26], appeared to be relevant to get this knowledge by *in virtuo* experiments.

Based on Bratman's work [3], the BDI (Belief, Desire, Intention) architecture is the most used to build rational and cognitive agents. A BDI agent has a set of beliefs (information about its environment), a set of desires (what it wants) and a set of intentions (actions to do to reach its goal) [20]. Its control process follows this algorithm: 1) to

perceive the environment, 2) to update beliefs and desires, 3) to choose an intention and 4) to act. Concerning emotions, since it is a recent subject of research, no consensus exists on a reference architecture as for the BDI model for rational agents. A first path of studies is the immersion of humans in a virtual world [5][25]. A second path, the one we have chosen to follow is the modeling of emotions for agents. Because of the nature of our application, we will focus on the second one. Several other projects investigate this path. The projects “Affective Reasoner” [15] and “PETEEI” [9] are interested in the social aspect of emotions without considering influence on reasoning. The “MRE” project is about emotions and planning but does not consider the notions of belief and desire. The BDE (Belief, Desire, Emotion) [12] agent is a BDI agent to which emotions are added. This method is the one we have chosen to build our new model of agent: the EFT (Emotion, Feeling, Temperament) agent.

This paper presents the work we realized to model and implement our prototype of simulation platform. We propose a model of emotional agents and of their interactions. This is the result of the collaboration between PerformanSe (skilled in behavioral tools), Aparta (implied in brain damaged people reintegration) and academic skills in computing and multi-agent systems. In the first part of this paper, we explain the context of the simulation and important notions of psychology, both general and specific to brain-damaged people, necessary to build a model. Then, we propose the model of our EFT agent based on the BDI model, emotional theories (OCC model [19]) and PerformanSe behavioral model. Then, we will present the modeling of the interactions of the system and their link with emotions thanks to the AgentUML formalism [18]. Finally, we will present the implementation of the model under the JADE platform [2] and the results we have obtained with this prototype.

2. Problem statement and preliminary concepts in psychology

2.1. Problem statement

PerformanSe already has 15 years of experience in designing tools for evaluating the behavior. It has also developed tools to determine *a priori* the dynamism of a

group in term of attraction / repulsion forces. However, no tools exists to simulate the evolution of the behavior of an agent and, even less so, of a group. Embedding the PerformanSe behavioral model into a virtual agent could be a way to obtain some results. We will see that the temperament is not sufficient to simulate the instant behavior of a person. Indeed, emotions have a first importance role in human reactions to environmental events. Thus, we also have investigated emotional models and the links between temperament and emotions. The goal is twofold: to bring together emotions and temperament to develop a convincing architecture of emotional agent, and to show the reusability of the PerformanSe model in a dynamic simulation.

2.2. Applicative framework

To construct a model of reusable agent adapted to any environment is quite an ambitious task. For the moment, to set the basis of our model, we have chosen to study a special case that will show the relevance of our simulation. The comparison between observed and simulated behavior in this environment will be a first step to validate our model.

The context of the simulation is a production line situated in a protected workshop, Aparta. In this company, the employees suffer from brain-damage after an accident and are here to learn how to reintegrate society. There are also paralyzed people, hemiplegic or paraplegic. Their mobility problems have to be taken into account in the simulation.

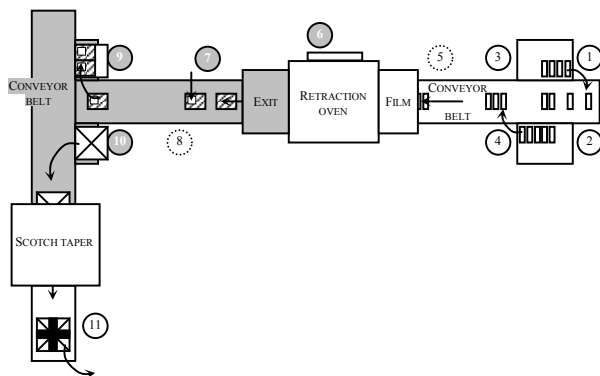


Figure 1. Packaging production line.

The modeled system is production line to repackage packs of cookies together (Figure 1). On this line, employees 1 and 4 put packs of cookies on the conveyor belt while 2 and 3 are supplying packs to 1 and 4. Then, these packs are wrapped in a film and go through an oven to be packaged in fours. Employee 6 is a non brain-damaged person and is responsible for the good functioning of the line. Moreover, he is there to encourage tired or unmotivated people. Then, at the end

of the oven, employee 7 puts a label on the package. Then, the packages arrive in front of employees 9 and 10 who put them into cartons for the transport. Finally, employee 11 closes the cartons and prepares them to leave the manufacture.

For our first implementation, we chose a subset of this line (the grey part) composed of employees 6, 7, 9 and 10. The choice is based on the variety of interactions contained by this subset. Indeed, we have interesting dependencies between people (the tasks of 9 and 10 depends on that of 7, besides people can talk together because of their nearness). The line, in this subset, also presents characteristic states that directly impact on the worker (state of a pack after the oven, speed of the conveyor belt). A subset enables us to implement a small application and make it more and more complex as systemic theory recommends it [17]. Moreover, validation and correction become easier.

Aparta is a particular company. It has the same kind of constraints of performance or efficiency as any firm but it also has to succeed in its reintegration mission. Discovering less stressing workstations on the line or groups with people getting on well together, for instance, is a major stake. Indeed, it improves both people's personal progress and productivity. It explains the interest of our simulation.

2.3. Specificities of brain-damaged people and emotional stake

The originality of this simulation is its application in an environment of brain-damaged workers. This context is very important since we are speaking of emotions, behaviors, human relationship. Therefore, we had to understand the effects of the most common injuries to the brain. The main specificity of a brain-damage lesion is that emotional filters disappear or are less efficient [16]. Moreover, brain-damaged people suffer from lapses of memory and concentration. The consequences of a brain damage show the importance of the dimension of emotions on their behavior. Besides, recent discoveries in neuropsychological studies [7] pointed out the necessity of an emotional process to be able to think and decide. This justifies the need we had to build convincing virtual agents with emotions.

This context is well adapted to build a sort of virtual humans. Indeed, a brain damage causes extreme behaviors such as inappropriate reactions, exaggerated demeanors. Therefore, the first step of a validation is to succeed in the realization of these spontaneous and unpredictable behaviors that are more reactive, because their emotional filters have disappeared, than those of non brain-damaged person.

2.4. Existing behavioral models

The human behavior is strongly influenced by emotions and the general psychological state of the person. It is the origin of non-rational reactions to an event, of individual specific behavior and partly of social interactions within a group. Brain damaged people are even more concerned due to their trauma. Therefore, to simulate human behavior, we have to consider two complementary aspects: emotion and personality.

We can distinguish four main types of emotional theories [6]: Darwinist, Jamesonian, cognitivist (appraisal theory) and social constructivist. The cognitivist approach is the most used in information technology because of a relatively simple implementation of this theory. The principle is to link the triggering of emotions with how the person appraises its environment. To draw a fast parallelism with the BDI model, emotions would be triggered after the beliefs of the agents are updated.

One of the most widespread cognitivist models of emotions is that of Ortony, Clore and Collins, often named OCC model [19]. They have sorted 22 emotions (11 pairs (Table 1)) in three main classes: those linked with the relevance of the consequences of an event compared to the expectations of the agent (most often its goals), those connected with the moral value of the actions of another, and those linked with the affection of an object. The model is especially designed for computer use. Therefore, most of the projects about emotional agents use this model with a few minor modifications [15][9][21]. In our case, this model is especially appropriate because the system is closed and the number of external events perceived by an agent is limited. Determining the emotional consequences of each event is then easier than in an open system where unpredictable external events could happen.

Table 1. The eleven pairs of the OCC model.

Happy-for ; Resentment
Gloating ; Pity
Joy ; Distress
Pride ; Shame
Admiration ; Reproach
Love ; Hate
Hope ; Fear
Satisfaction ; Fear-confirmed
Relief ; Disappointment
Gratification ; Remorse
Gratitude ; Anger

For the behavioral aspect, we have worked on the model of the PerformanSe company [13]. It is based on the systemic theory developed in the fifties by the School of Palo Alto [11]. This group of researcher defined the communication as relational and interactional processes.

Relations between people are more important than people themselves. Therefore, each behavior is seen as a communication act. The set of relations is considered as an interacting communication system. A disrupted communication between two people denotes behavioral trouble.

The PerformanSe model has followed this theory considering the person as a system interacting with a context (environment, family and working circles, working rules, etc.) [13]. This system is defined by ten behavioral dimensions (Table 2) interacting with each other and with the context presented. Each dimension is balanced between two bipolar personality traits (ex. Combative / Conciliation). This model is an extension of the "Big Five" [11] model which has concluded that personality traits are linked with the behavior and that they could be represented in five dimensions (assertiveness, benevolence, sensibility, open-mindedness, sense of responsibility). Three motivations complete this model and reflect the major trends a person would follow in his/her behavior. Finally, bipolarity also introduces the potential of a person and its variability zone in a dimension. For example, somebody could be combative, but with a large variability zone and a potential oriented towards conciliation. It means that, if needed, it could become more conciliatory.

Table 2. The ten behavioral dimensions.

Extroversion ; Introversion
Anxiety ; Relaxation
Assertiveness ; Questioning
Receptiveness ; Determination
Rigor ; Improvisation
Intellectual dynamism ; Intellectual conformism
Combative / Conciliation
Motivation of achievement ; Motivation of facilitation
Motivation of belonging ; Motivation of independence
Motivation of power ; Motivation of protection

3. Modeling of the EFT agent

3.1. The need of introducing feelings

Our emotional agent is built on the BDI model. It is structured in three parts: its personal features, its memory and its procedural system (Figure 2).

Personal features stand for the set of inner behavioral characteristics of the agent. First, the originality of our model is the agent personality divided into three levels: emotions, feelings and temperament. We based this separation on the variability of each level. Emotions are ephemeral, representing instant feelings. For instance, you only feel frightened for a relatively short time (the duration depending both on the intensity and the

temperament of the person). Concerning temperament, it represents the nature of our agent. It is supposed to be constant in time. Indeed, if we consider “extroversion”, we assume this feature does not evolve. Just keeping these two parts did not enable our agent to remember its interactions or what it felt. For example, an agent is more likely to become friendly with an agent that gives him happiness than with another. That is why we introduced the notion of the feelings (friendship, for example). Feelings evolve with the emotions felt and in function of the agent's temperament.

3.2. Description of the EFT part of our model

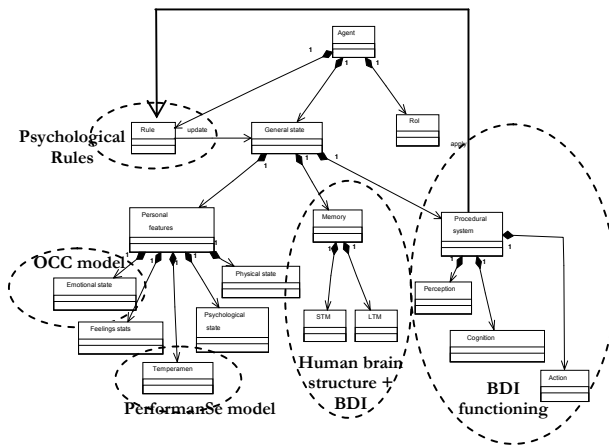


Figure 2. UML class diagram of the EFT agent.

To build the personality of an agent, we used proven models. Emotions are based on the OCC model as described above with its 11 pairs of emotions (Table 1). We complete this model by conditioning the generation and the intensity of an emotion by the temperament. Moreover, we split emotions in two classes: personal emotions which do not imply someone else (pride, shame for instance) and relative emotions which are linked to another agent (compassion or taste). For instance, somebody who is rather anxious and extroverted will be angry more easily than an introverted and relaxed one. To model the temperament, PerformanSe behavioral model is used. Since the temperament does not evolve in our simulation, it will always be used as a conditional parameter of psychological rules. Relating to feelings, we did not find any model.

Therefore, we worked with psychologists to develop a model of six pairs of feelings presented in Table 3. These feelings can also be divided in the personal and relative feelings. This notion is interesting because it enables each agent to have its own personality (personal feelings and emotions) but it can also have several types of relationships with the other agents (relative feelings and

emotions). Emotions and feelings relative to someone else are part of the social knowledge of an agent. Indeed, it corresponds to a part of the beliefs of the agent that will determine its social relations. The difference with usual beliefs of the BDI model is that those are not the result of a process of the agent to apprehend its environment. They are only the result of its social interaction with other agents. Another point is that feelings and emotions affect the way the agent and chooses its goals and intentions in a non-rational way contrary to the BDI model.

Table 3. Six pairs of feelings for the EFT agent.

Quietude	Worry
Self respect	Low regard
Friendship	Hostility
Trust	Distrust
Indebtedness	Rancor
Compassion	Indifference

3.3. Physical and psychological dimensions

Table 4. The psychological dimension.

Experience
Stress
Instability
Disturbance under unusual events
Mental fatigue
Motivation
Negligence

This three-fold personality (emotions, feelings and temperament) is completed with psychological and physical dimension. The first one concerns the internal state of the agent regarding its work. This aspect has already been experimented in an earlier simulation prototype. Psychological experts from PerformanSe and supervisors from Aparta have validated the choice of the variables proposed in Table 3. In this simulation, their evolution obviously depends on the emotional state of the agent. Finally, physical characteristics enable to retain the spatial position of the agent and its expression. For the face, we used Ekman's works about emotions and face representation [8]. In the case of disabled people, it can also model physical handicaps that then could affect the psychological state of the agent. For instance, somebody in wheelchair could emotionally suffer from his/her inability to perform some actions.

3.4. The memory of the agent

Regarding memory, we separate two aspects based on psychological characteristics: the long-term and the short-term memories. The long-term memory stocks information concerning learning and knowledge about

working. This memory is rather static contrary to the short-term memory. Indeed, the short-term memory is the memory where the agent keeps information about its environment (agents and objects), itself (internal state) – *i.e.* its beliefs –, its desires and its intentions. This memory is built on BDI architecture. It is very dynamic since environment is very dynamic. Concerning other agents, an agent knows what the others seem to be. Indeed, an agent cannot guess an internal state but just suppose it thanks to the expression or the physical state of the other agents. Therefore, we make a distinction between the real internal state of an agent and the state the other agents believe it has. Temperament is again very important in the expression of internal state. For instance, an extroverted person will be more expressive than an introverted one.

Concerning desires, we propose two types of desires. Passive desires are desires that do not depend on the agent action but on the global evolution of the environment which can trigger expected or dreaded events. On the contrary, active desires depend on the agent itself. We use this division in the choice of actions. Indeed, an agent can only try to realize its active desires. The passive desires are just used in the evaluation of an event (expected or not) for the generation of emotions. Consequently, the intention of an agent is the active desire which it most expects. This intention is linked to a plan of elementary actions and the goal of an agent is the current action of the plan.

3.5. Control process of the agent

The procedural system corresponds to the cognitive functioning of the agent and is based on BDI architecture. First, the agent perceives its environment. The perception is filtered according to the personal state (a relaxed agent will be less sensitive to malicious remarks) and physical state (an agent with a little deafness will less perceive a loud noise). Then, the agent updates its beliefs in consequence of the filtered message, decides what to do and then acts.

Finally, a base of psychological rules (established with experts) links all these parts. These rules manage personality evolution and influence, choice of actions and kind of communications. For instance, rules can be expressed in the following manner: “If I am very stressed and very anxious and angry and my neighbor is speaking to me harshly, therefore I will shout at him”. We distinguish ten groups of rules given in Table 5. This categorization enables us to list all the needed rules and to gradually build the simulation adding the rules in each group progressively. A future objective is also to enable a non-computer expert to add rules to the system. Then this type of classification is needed to do that.

Table 5. Classification of psychological rules.

Rules	Role
Group 1	Filter speaking messages (psychological filters).
Group 2	Filter messages from the line (mechanic filters).
Group 3	Update after speaking messages.
Group 4	Update after messages from the line.
Group 5	Update after the realization of a goal.
Group 6	Generate working messages.
Group 7	Generate relationship messages.
Group 8	Generate answer messages.
Group 9	Generate a goal / intention.
Group 10	Internal evolution of parameters.

The control process of the agent is its functioning. It enables the agent to act, talk or react. Talking and acting are parallel processes. In our case, communication is not a means for the agent to collaborate or to negotiate with other agents. We only use communication as a vector to have our agents exchange their feelings. In this way, we cannot really consider exchanged messages as speech act [1][24], they are not a means for the agent to achieve its goals. It is the reason why we have separated the two processes. This choice simplifies protocol management and thus, an agent can act and speak at the same time, which increases its reactivity. In this part, we will just describe the control of acting (Figure 3) since speaking is a very simple process (get the message, update the personal state and generate possible answers).

First, all the perceptions (messages) of an agent are stocked in a buffer and analyzed one by one. The first message of the buffer is analyzed (filter and update of beliefs, desires and internal state). Then, the strongest desire becomes the intention of the agent. At this moment, it checks if its current goal is still in adequation with its intention. These two steps are important because they enable an agent to change its intention before completing it (the change is between two elementary actions of the plan). We talk about a BDI agent with open-minded commitment [20] – *i.e.* it will maintain an intention as long as it is still believed possible. Thus, the agent is very reactive to the environment evolution. After that, if the goal corresponds to the intention, the agent acts. Otherwise, it changes its goal to fit its intention.

We can see in Figure 3 that in plan search the agent has the ability to construct individual plans and collaborative plans to reach its goal. We do not yet work on this part which implies planning and requires a lot of investigations to have emotions impact on it.

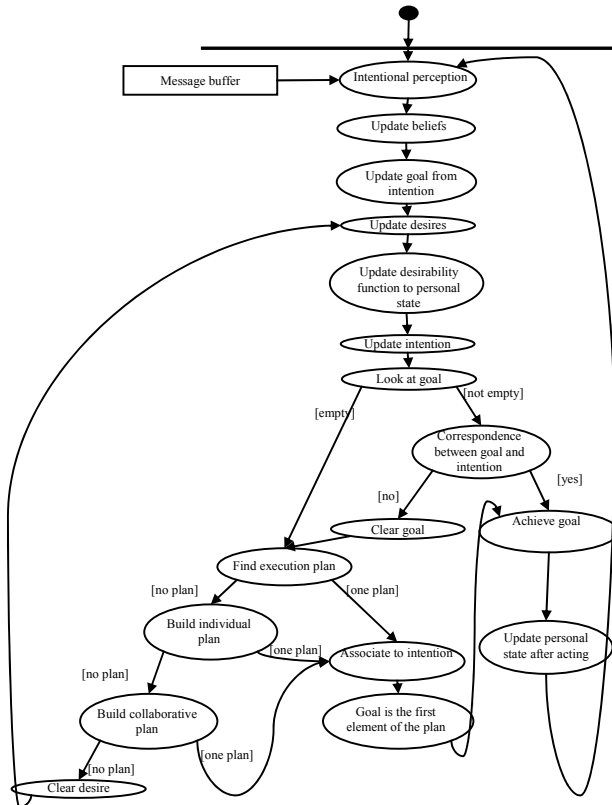


Figure 3. Control process of the EFT agent.

We will now insist on how emotions and more generally how internal state take place in this cognitive architecture. The main influence is on reasoning. Indeed, emotional variables affect the action chosen by the agent. Following its personal state, an agent will not have the same choice of possible actions. On the other hand, relationships are memorized thanks to feelings. Therefore communication is also influenced by the internal state of an agent. Temperament is necessary to enable each agent to react differently to the same event. Finally, emotional state is strongly linked to the physical state which enhances the possible relational situations.

4. Modeling of the system using Agent UML

The work around AgentUML began with the difficulties to model multi-agents systems only with UML [18]. AgentUML consists in adapting UML diagrams to oriented agent programming. The main propositions concern interactions diagrams. The conditional structures are simpler and modeling communication between agents is easier. Moreover, AgentUML introduces the notion of interaction protocols which give a solution to formalize exchanged messages and are reusable. That is why, after using UML for the agent, we have used AgentUML to model the system.

4.1. Interactions of the system

The interaction aspect of an MAS is crucial. Indeed, *“there's no such a thing as a single multi-agent system”* [26]. Interactions and agent communication are the basis of the emergence of group behavior, particularly interesting in our case. What we are interested in is seeing the evolution of someone interacting with the machine and with its co-workers. Thus, it becomes necessary to describe the set of primary interactions that leads a person to evolve. To achieve this goal, we have both worked with psychologists (expert of brain-damaged people) and observed the behavior of the workers on the line.

We have followed a three-step top-down methodology. First, we have designed a global overview of the system. We have identified the entities of the system and discovered the set of their interacting relations with UML use-case diagrams (Figure 4). The second step consists in describing the activity of the system with AgentUML activities diagrams [18]. Two types of diagrams have been realized: a global activities diagram on the overall normal running of the system and more specific diagrams on each workstation (each agent) of the assembly line. Finally, we have detailed each of the previous activities diagrams with AgentUML sequence diagram.

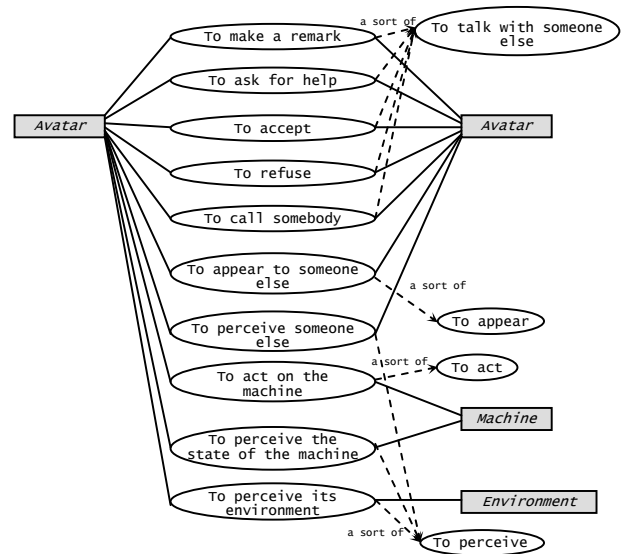


Figure 4. General UML use-case diagram.

We have chosen a number of key interactions of the system, which are characteristic of the modeled environment and source of the dynamism of social relations. Our simulation does not focus on language; we privilege the reactive aspect and its impact on emotions, to the dialog. We have distinguished four classes of interactions:

- to communicate with somebody,

- to appear to somebody,
- to act physically on the environment,
- to perceive its environment.

We want to precise that these classes do not come as some new FIPA-ACL performatives [10]. Here, we have described the most crucial interactions of the system we study to observe characteristic behaviors. We then conformed to the FIPA-ACL norm using the right performatives for our messages. We do not need, at this stage, to have rich content in our messages (with ontologies or logic formalism), first because in our context the messages exchanged by workers are not very rich and second because we do not have rich collaboration or negotiation processes between people.

Communication between people has an important part in the emotional dynamism of the system. Emotions influence the way people talk to each other both in the wording and in the manner the sentence is expressed – tone, velocity, loudness. The question of semantic analysis to extract the good/ill will of a message sent by an agent to another is beyond our scope and is a quite hard research subject. To consider this concept, we have simply introduced an intention coefficient varying from zero to one that translates the malevolence/benevolence of a message (Figure 5). Moreover, a remark could be positive (compliment) or negative (reproach). Thus, we have distinguished these two types of remark.

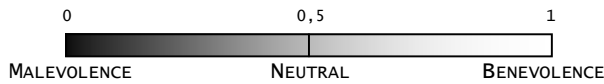


Figure 5. Malevolence/Benevolence coefficient.

4.2. Protocols of interaction

We have first designed an activity diagram to describe the general production line functioning (Figure 6), and specific activity diagrams for each workstation of the line (workstation 7 in these examples).

We have then designed general protocols shared by all the human agents of the system. The “voluntary perception” protocol describes the fact that an agent decides to perceive another agent (human or machine) while it is true that in an environment, you also receive unsolicited stimuli (like noise, light). It is particularly important in our case to make this distinction because emotional state of brain-damaged people is strongly affected by unsolicited stimuli due to their weak capacity of concentration. Moreover, a workshop conveys a lot of perturbation (noise of the machine, excessive or insufficient light for instance). The general “working protocol” describes how an agent interacts on the production line. For each workstation of the line, we have realized a sequence diagram of the working protocol describing precisely sequence of the tasks to accomplish.

The last and richest type of interaction is the communication between people. We can distinguish “remark”, “help request” and “inform” protocols.

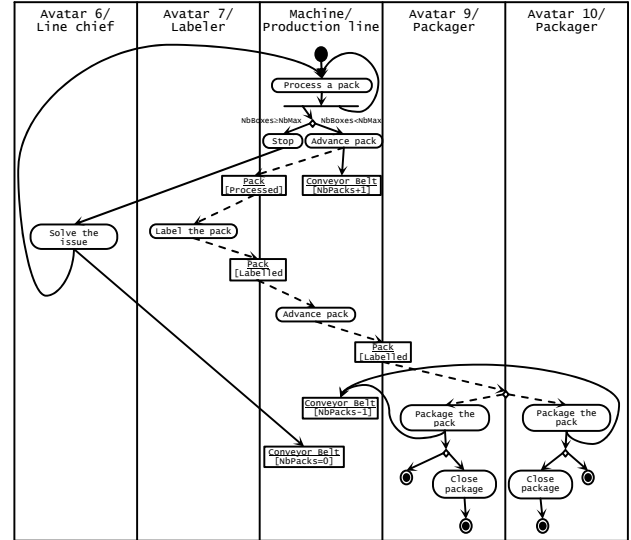


Figure 6. General UML activities diagram.

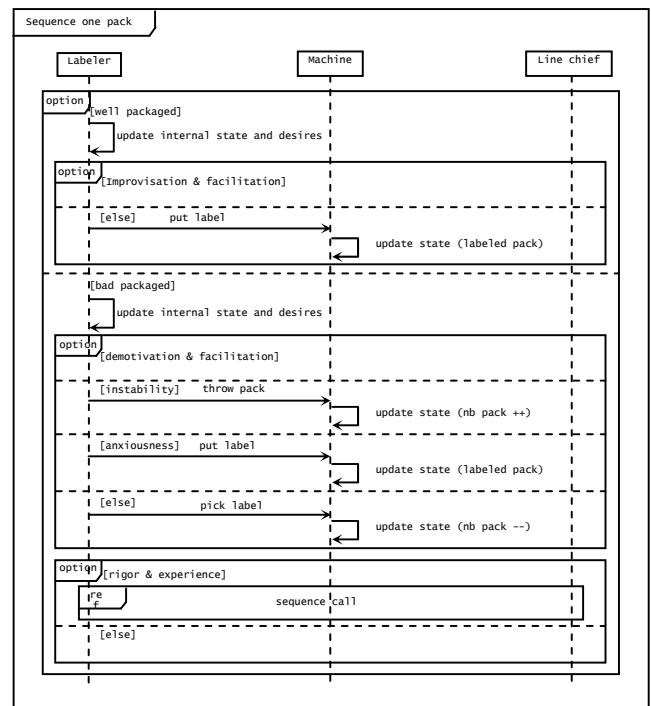


Figure 7. AgentUML sequence diagram for workstation7 (sequence one pack).

Finally, we have designed precise sequence diagrams for each workstation of the line and for each situation the worker is confronted with. For instance, the labeler can encounter three cases in its work zone on the conveyor belt: either there is no pack, or there is one pack (normal

case), or there are too many packs (jamming). In each of these cases, we have to describe how the person reacts according to his/her emotional state. If the line is jammed, and the person is stressed, anxious and angry, (s)he will throw packs. AgentUML notations enable us to represent such multi-conditional structures quite easily as shown on Figure 7.

5. Realization

5.1. The prototype of the implemented simulation

First, we have chosen to implement our simulation under the JADE platform [2]. The first reason is the existence of an earlier prototype wrote in Java language. Thus, it was logical to base our work on this previous simulation and, then, to use a Java-powered platform. JADE is also really easy to use and one can rapidly develop agents thanks to the mechanisms proposed by the platform. Open-sourced and widely used both in academic and industrial projects, JADE has already proved its value and is abundantly documented. Moreover, it is not only a development library but also an execution tool which gives great abilities to trace and control its agents. Finally, this platform is widely open and numerous plugins and third-party contributions already exist. To give but one example, JadeX [4] is a recent extension that makes easier the development of agents based on the BDI architecture.

At the first stage of this project, only a part of the model proposed was implemented. Due to expert's availability constraints, the psychological expert evaluation was only available for a few of the parameters. The choice of these parameters was made with experts to enable us to simulate simplified behavioral patterns. Three classes of comportment have been defined: paranoid, well-adapted and ill-adapted. Each one is described with the three personality traits: extroversion, anxiety and combativeness. It contributes to the validation of the prototype because these behaviors are quite easy to detect on a simulation by looking at the evolution graphs of emotional parameters. Reproducing these behaviors partly proves the value of the model and of the rules.

We present on Figures 8 and 9 two screenshots of the simulation. The main screen (Figure 8) shows the line and the agents working. An agent is represented by a "Smiley" since it is an easy and expressive way to illustrate simple emotions. There is a special agent with a cap on its head, it is the supervisor and, here, it is not an EFT agent. It acts as a reactive agent by consoling workers in difficulty and repairing the line when there is a problem. Speaking messages are represented in speech balloons. Thus, the user regarding the illustrative vocabulary of the sentence can guess the intention of a remark. Then, each agent has a window (Figure 9) to

represent its internal state. In the top left corner, the temperament is drawn and the graphs represent the evolution of the variables. We propose this screen to give the user an idea about the possibilities linked with such a simulation (correlations between variables, links between a situation of crisis observed on the main screen and one or more parameters).

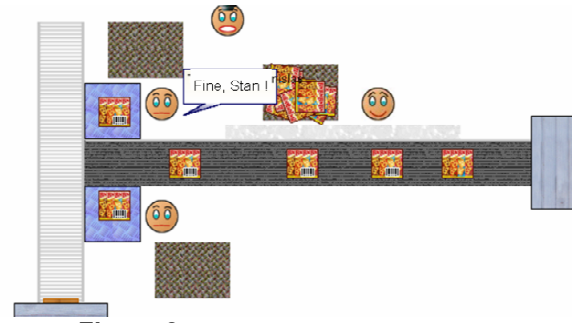


Figure 8. Screenshot of the main screen.

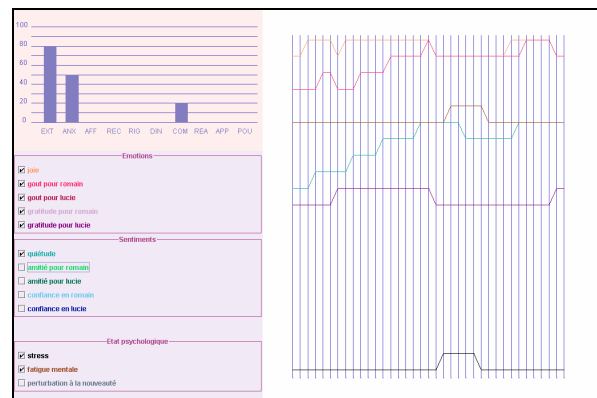


Figure 9. Screenshot of the internal state of an agent.

5.2. Results and expectations

This application gives promising results. We have reached our first goal that was to observe different behaviors for different initializations of the agents. For instance, a much-stressed paranoid agent will throw unlabelled packs whereas a conscientious agent will put them away. Moreover, on the larger scale of the overall system, we have observed there were differences both in terms of work efficiency and psychological "well-being" depending on the composition of the group of agents. The simulation has already led to modifications on the line to favor the rotation of workers at a much-stressing workplace.

Now, the first thing to envision is to get much more psychological expert evaluation and then to add new possible interactions between agents and with the environment. The main difficulty here is that the number of emotional rules to collect increases with the number of

the possible actions of the agent. Indeed, each action must have its own triggering rules. At this point, we could envision two solutions: determining some meta-rules for meta-classes of actions or thinking to a less determinist system that does not need this kind of rules. In all cases, it is essential to make a simulation more exhaustive and consequently more reliable for the deciders. We will have to work with them to evaluate and satisfy their new requirements that will appear along with an improvement of the simulation.

6. Conclusion

In this paper, with the idea to build a decision-support application to encourage the reintegration of brain-damaged people, we have proposed an emotional agent model. The notion of emotion was developed along three axes: emotions, feelings and temperament (EFT agent). The distinction of these three axes based on the duration of the evolution of each dimension is the keystone of our model. More important than the emotions, feelings and temperament parameters proposed, the idea of splitting the emotional aspect into multiple dimensions can be a good basis to develop convincing virtual humans. Then, in our model, the psychological aspect plays a role at different levels of the control process: cognitive, perceptive and expressive. It is an important stake to be able to simulate all the aspects of emotions on the behavior.

In the short term, our objective is to complete psychological appraisal to enhance the simulation and, consequently, the complexity of potential behaviors. Then, we will have to study emergent behaviors and compare them to real-life behaviors already observed. Then, it will be interesting to extend the study to non-brain-damaged people. We have built a model that can take into account the specificities of brain-damaged people, but that is also sufficiently generic to start applying the same kind of approach to non brain-damaged people. In a second time, it would also be interesting to adopt a methodology to evaluate our emotional model as done in [14].

7. References

- [1] J. L. Austin, *How to do things with words*, Oxford University Press, 1962.
- [2] F. Bellifemine, A. Poggi, G. Rimassa, JADE-A FIPA-compliant agent framework, in *Proc. Of the PAAM 99*, 1999, pp. 97-108.
- [3] M. Bratman, *Intentions, Plans and Practical Reason*, Harvard University Press, 1987.
- [4] L. Braubach, A. Pokahr, D. Moldt, W. Lamersdorf, Goal representation for BDI agent systems, in *The second international workshop on Programming Multiagent Systems (PROMAS 04)*, 2004.
- [5] A. Camurri, A. Coglio, An architecture for emotional agents, *IEEE Multimedia*, 1998, 5(4):24-23.
- [6] R. R. Cornelius. Theoretical approaches to emotion, in *SpeechEmotion-2000*, 2000, pp. 3-10.
- [7] A. Damasio, *Descartes' error: emotion, reason and the human brain*, Putnam Books, 1994.
- [8] P. Ekman, W. Friesen, P. Ellsworth, *Emotion in the human face*, Cambridge University Press, 1972.
- [9] M. S. El-Nasr, T. R. Ioerger, J. Yen, A web of emotions, in *Proc. Of the workshop on Emotion-Based Agent Architecture (EBAA 99) at the 3rd International Conference On Autonomous Agents (Agents 99)*, 2000.
- [10] FIPA, FIPA ACL Message Structure Specification, FIPA Specifications, <http://www.fipa.org/specs>, 2000.
- [11] D. W. Fiske, Consistency of the factorial structures of personality ratings from different sources, *Journal of abnormal and social psychology*, 1949, 44:329-344.
- [12] A. M. Florea, E. Kalisz, Behavior anticipation based on beliefs, desires and emotions, in *Proc. Of the 6th international conference on Computing Anticipatory Systems (CASYS 03)*, 2003.
- [13] R. Gras, P. Peter, S. Baquedano, J. Philippé, Structuration de comportements de réponse à un questionnaire par des méthodes multi-dimensionnelles, *Extraction et gestion des connaissances, extraction des connaissances et apprentissage*, 2003, 17(1-3) :105-118.
- [14] J. Gratch, S. Marsella. Evaluating the modeling and use of emotion in virtual humans, in *Proc. Of Autonomous Agents and Multi-Agent Systems (AAMAS 04)*, 2004.
- [15] A. Kapoor, S. Mota, R. W. Picard, Towards a learning companion that recognizes affect, in *Proc. Of emotional and intelligent : the tangled knot of social cognition, AAAI Fall symposium*, 2001.
- [16] M.V. Der Linden, X. Seron, DL Gall, P. Andrès, *Neuropsychologie des lobes frontaux*, Solal, 1999.
- [17] J. Méléze, *L'analyse modulaire des systèmes de gestion*, A.M.S, Hommes et techniques Ed., 1972.
- [18] J. Odell, H. Van Dyke Parunak, B. Bauer, Extending UML for agents, in *Proc. Of the agent-oriented information systems workshop at the 17th National Conference on Artificial Intelligence (NCAI 00)*, 2000, pp. 3-17.

- [19] A. Orthony, G. Clore, A. Collins, *The cognitive structure of emotions*, Cambridge University Press, 1988.
- [20] A. Rao, M. Georgeff, Modeling rational agents within a BDI-architecture, in *Proc. Of the 2nd International Conference on Principles of Knowledge Representation and Reasoning (KR 91)*, Morgan Kaufmann Publishers Inc, 1991, pp. 473-484.
- [21] W. S. Reilly, *Believable social and emotional agents*, PhD Thesis, Carnegie Mellon university, Pittsburgh, 1996.
- [22] J. Rickel, S. Marsella, J. Gratch, R. Hill, D. Traum, W. Swartout, Steve goes to Bosnia : towards a new generation of virtual humans for interactive experiences, *IEEE Intelligent Systems*, 2002, 17(4):32-38.
- [23] V. Sandoval, *L'informatique décisionnelle*, Hermès Sciences, 1997.
- [24] J. R. Searle, *Speech acts: an essay in philosophy of language*, Cambridge University Press, 1969.
- [25] J. Velasquez, Modeling emotions and other motivations in synthetic agents, in *Proc. Of the 4th International Workshop on Agent Theories, Architecture and Languages (ATAL 97)*, 1997.
- [26] M. Wooldridge, *An introduction to multi-agents systems*, Wiley, 2001.

UN SYSTEME D'AGENTS EMOTIONNELS POUR L'AIDE A LA DECISION

Stéphane Daviet, Hélène Desmier, Gaële Rozec

Equipe Connaissance Optimisation Décision, LINA, Site EPUN - PerformSe SAS

Résumé

Dans le cadre du développement d'un outil d'aide à la décision pour la réinsertion d'individus cérébrolésés dans le milieu ordinaire du travail, nous avons développé une plate-forme de simulation des comportements, basée sur les systèmes multi-agents. Pour cela, nous avons dû, d'une part, élaborer un modèle d'agent prenant en compte la dimension émotionnelle dans le comportement humain. Le modèle EST (Emotion, Sentiment, Tempérament), présenté ici, est une extension du modèle cognitif rationnel BDI. D'autre part, nous avons également caractérisé puis formalisé les interactions clés du système. Cet outil doit permettre d'observer l'émergence de comportements dans un contexte de travail.

Mots clefs: aide à la décision, agent émotionnel, systèmes multi-agents

1 Introduction

Traditionnellement, en Informatique Décisionnelle, les outils développés permettent l'analyse des données archivées par une entreprise afin d'en extraire des connaissances pertinentes [Sandoval, 1997]. Dans le cadre du projet TC&Plus.Virtuel, notre objectif est de proposer un outil d'aide à la décision pour une entreprise dont le rôle est d'accompagner des personnes cérébrolésées en vue de leur réinsertion dans le milieu ordinaire du travail. Il s'agit donc ici d'obtenir des connaissances sur le comportement d'individus au travail. Le recueil de toutes les données observables

étant évidemment très délicat, la simulation informatique, basée sur un système multi-agents [Wooldridge, 2002], est apparue comme un moyen pertinent pour les obtenir. Dans ce cadre, nous avons été amenés à étendre le modèle d’agent rationnel classique [Bratman, 1987] afin de prendre en compte la dimension émotionnelle essentielle chez les personnes cérébrolésées, dépourvues d’une partie de leurs filtres émotionnels [der Linden et al., 1999], et plus généralement dans le comportement humain [Damasio, 1994].

Notre simulation repose d’une part sur un modèle individuel d’agent, et d’autre part sur les interactions entre individus au sein du système multi-agents. Dans cet article, nous allons donc présenter le modèle d’agent émotionnel que nous avons développé. Nous nous sommes appuyés sur l’architecture d’agent rationnel BDI (Belief, Desire, Intention) [Rao and Georgeff, 1991] et sur les modèles OCC (Orthony, Clore et Collins) [Orthony et al., 1988] et PerformanSe [Gras et al., 2003] respectivement pour les émotions et le tempérament. Nous avons opté pour une structuration de la dimension émotionnelle de notre agent selon trois axes : émotion, sentiment et tempérament. Nous détaillerons également le travail effectué au niveau de la caractérisation et de la modélisation des interactions.

Nous présenterons d’abord un bref état de l’art des modèles cognitifs et des travaux relatifs à la modélisation des émotions de l’individu. Nous détaillerons ensuite le travail réalisé : le modèle d’agent que nous avons construit et sa formalisation en UML (Unified Modelling Language). Enfin, nous présenterons l’aspect multi-agents et les résultats obtenus.

2 De l’agent rationnel à l’agent émotionnel

Basée sur les travaux fondateurs de Bratman [Bratman, 1987], une des architectures les plus utilisées actuellement pour implémenter un agent cognitif rationnel, est l’approche BDI [Rao and Georgeff, 1991]. Dans cette architecture, un agent

possède des croyances, qui représentent toutes les informations qu’il a sur son environnement (y compris sur les autres agents). Ses désirs sont les états de l’environnement qu’il souhaiterait atteindre et ses intentions sont les actions qu’il va effectuer pour atteindre ses désirs. Le processus de contrôle est défini par l’algorithme suivant : percevoir son environnement, mettre à jour ses croyances, mettre à jour ses désirs, choisir son intention, lier à celle-ci un plan d’actions et enfin l’exécuter [Wooldridge, 2002]. L’intégration de la dimension émotionnelle dans ces agents est beaucoup plus récente et s’inspire des recherches de la dernière décennie en psychologie sur le rôle des émotions dans les processus de raisonnement [Damasio, 1994]. Deux voies complémentaires sont explorées : l’une consiste à immerger un humain dans un univers virtuel ; l’autre, sur laquelle nous nous focalisons ici, porte sur la modélisation d’émotions dans des agents virtuels, qualifiés alors d’émotionnels [Orthony et al., 1988]. Différents projets récents s’inscrivent dans cette démarche. Les projets “Affective Reasoner” [Kapoor et al., 2001] et “PETEEI” [El-Nasr et al., 2000] prennent en compte l’aspect social des émotions mais insistent peu sur l’influence dans le raisonnement. Le projet “MRE” [Rickel et al., 2002] porte sur les émotions dans la planification mais ne s’intéresse pas aux désirs et croyances. Le modèle BDE [Florea and Kalisz, 2003] (Belief, Desire, Emotion) étend le modèle de l’agent rationnel BDI, cité précédemment, en lui greffant des émotions. La spécificité applicative de notre problématique nous a conduit à suivre une démarche similaire mais orientée ici vers un objectif de simulation et non d’immersion comme dans le modèle BDE d’origine.

3 L’agent EST (Emotion, Sentiment, Tempérament)

3.1 Modélisation

Notre modèle s’appuie sur une structuration en triptyque (Emotion, Sentiment, Tempérament) qui permet de prendre en compte les différentes échelles temporelles

d'évolution de ces paramètres. La dimension Emotion suit le schéma du modèle OCC [Orthony et al., 1988]. Elle est donc constituée de onze couples d'émotions (joie/tristesse, goût/dégoût, par exemple) exprimant le ressenti de l'instant en réaction à un évènement. Un paramètre émotionnel peut rapidement évoluer mais sa persistance dans le temps est faible. La dimension Tempérament représente le caractère de l'agent. Il est construit sur le modèle PerformanSe [Gras et al., 2003] qui décrit un individu comme un système en interaction avec son environnement. Ce système est un ensemble de dix traits de personnalité : sept dimensions comportementales (extraversion/introversion, anxiété/détente, combativité/conciliation, réceptivité/détermination, affirmation/remise en cause, dynamisme/conformisme intellectuel, rigueur/information) et trois motivations (réalisation/facilitation, appartenance/indépendance, pouvoir/protection). Son évolution temporelle est faible et considérée ici comme un paramètre constant. Dans notre application, les deux dimensions précitées ne nous permettaient pas d'intégrer l'impact à long terme des émotions. Ceci nous a conduit à l'ajout de la dimension Sentiment qui, à notre connaissance, a été peu explicitement considérée dans la littérature multi-agents. Cette dimension est composée de six couples de variables, choisis avec des psychologues, que sont l'amitié/inimitié, la confiance/méfiance, la reconnaissance/rancune, la compassion/indifférence, la quiétude/inquiétude et l'estime/més-estime de soi. Les sentiments se construisent en fonction des émotions ressenties et du tempérament. L'évolution temporelle est donc progressive et plus lente que celle des émotions. La structuration de la mémoire d'un agent EST s'organise selon une dichotomie grossière : long terme et court terme. La mémoire à long terme est composée d'une bibliothèque d'actions élémentaires et de plans d'actions correspondant aux savoir-faire et aux apprentissages de l'agent. Son évolution est plus lente que la mémoire à court terme dans laquelle l'agent retient toutes les informations sur son environnement et son état interne, structurées sur le modèle BDI en croyances, désirs (dépendants ou non de son action) et intention (désir de plus forte désirabilité). Le fonctionnement de l'agent est basé sur un système procédural de type BDI où chaque étape est conditionnée par l'état interne de l'agent (choix

de l'action, du message, interprétation d'un évènement, évaluation des désirs). Ces parties évoluent et interagissent ensemble grâce à une base de règles qui est, en fait, la formalisation de l'expertise recueillie.

3.2 Formalisation UML

Notre formalisation d'agent EST s'appuie sur le langage UML (Unified Modelling Language) [OMG, 2003]. Le choix de ce formalisme est justifié par sa généricité et son universalité dans le domaine objet, sa précision due à l'existence d'un métamodèle et son intelligibilité par des non-spécialistes. Le diagramme de classes (figure 1) formalise la modélisation en quatre grandes parties : une base de règles, des caractéristiques personnelles (composées du triptyque Emotion, Sentiment, Tempérament), une mémoire (composée de mémoires à long et court terme) et un système procédural. Le processus de contrôle d'un agent EST est lui formalisé en un diagramme d'activités (figure 2). Il détaille toutes les étapes du fonctionnement cognitif de l'agent EST. Tout d'abord, la gestion des messages et des actions est effectuée parallèlement. Nous allons nous intéresser ici à la gestion des actions. L'agent perçoit intentionnellement son environnement afin de mettre à jour ses croyances. A tout moment, l'agent a une intention courante, qui est ce qu'il est en train de faire, divisée en actions élémentaires. Après avoir mis ses croyances à jour, l'agent met son objectif (action élémentaire à exécuter) à jour en passant à l'étape suivante de son intention. Puis, il met à jour ses désirs et choisit comme intention courante le désir de plus forte désirabilité. A cet instant du processus, il vérifie que son objectif est toujours en adéquation avec l'intention courante. Cette vérification permet à l'agent de changer d'intention avant de l'avoir terminée ce qui lui assure une bonne réactivité. On parle alors d'agent à obligation ouverte [Wooldridge, 2002]. Si l'objectif est en adéquation avec l'intention, l'agent effectue son action. Sinon, il efface son objectif et recherche le nouveau qui va correspondre à sa nouvelle intention.

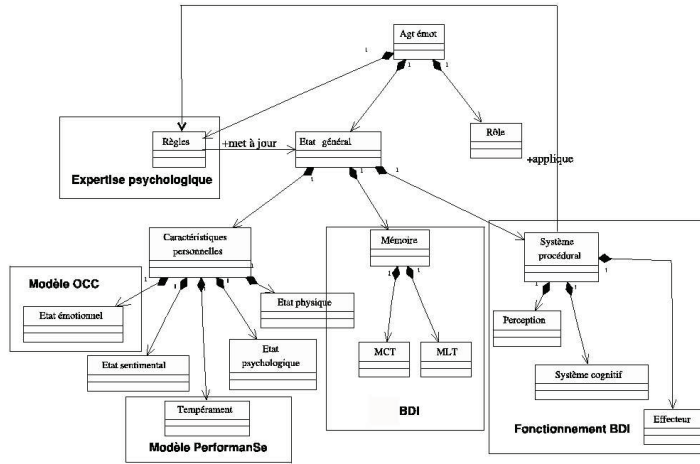


FIG. 1. Diagramme des classes UML d'un agent EST, le losange noir traduit un lien de composant/composé.

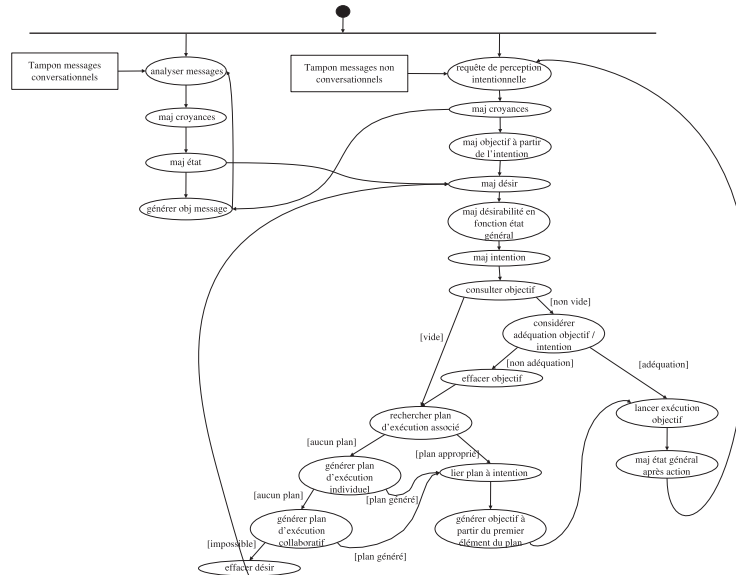


FIG. 2. Diagramme d'activités UML du processus de contrôle de l'agent.

4 La plate-forme multi-agents

Nous avons implémenté une plate-forme de simulation, dans l'environnement multi-agents JADE [Bellifemine et al., 1999]. Elle met en scène quatre agents travaillant sur une chaîne de conditionnement de paquets (figure 3). Pour cette première réali-

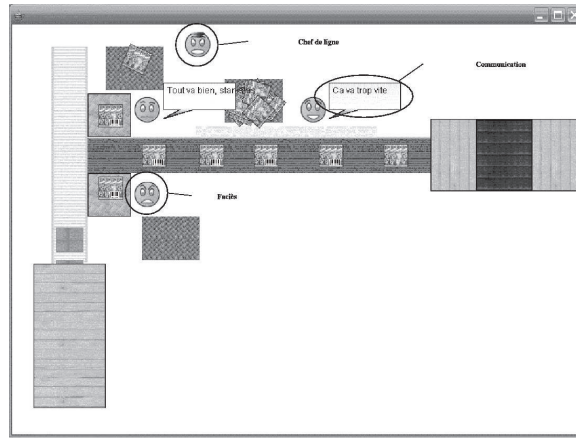


FIG. 3. Capture d'écran de la simulation.

sation, nous avons opté pour des profils d'agents caricaturaux et une représentation du milieu de travail simplifiée.

5 Conclusion

Dans cet article, guidés par une application d'aide à la décision dans le cadre de l'aide à la réinsertion de personnes cérébrolésées dans le milieu ordinaire du travail, nous avons proposé un modèle d'agent intégrant la dimension émotionnelle de l'individu. Cette dimension a été développée selon trois axes : émotions, sentiments et tempérament. Il était également important que cette dimension intervienne à la fois au niveau cognitif, perceptif et expressif dans le processus de contrôle de notre agent. Une forme simplifiée du modèle a été implémentée au sein d'une plate-forme expérimentale.

A court terme, l'objectif est de compléter l'expertise psychologique afin d'enrichir la simulation et la complexité des comportements observables. Une expérimentation devra ensuite être réalisée afin de dresser une typologie des comportements émergents et de les comparer aux comportements réellement observés. D'autre part, notre modèle étant suffisamment générique, une réflexion est en cours pour appliquer cette approche à des individus ne souffrant pas d'un handicap.

Références

- [Bellifemine et al., 1999] Bellifemine, F., Poggi, A., and Rinassa, G. (1999). Jade - a fipa-compliant agent framework. In *Proc. of the PAAM 99*, pages 97–108.
- [Bratman, 1987] Bratman, M. (1987). *Intentions, Plans, and Practical Reason*. Harvard University Press.
- [Damasio, 1994] Damasio, A. (1994). *Descartes' error : emotion, reason, and the human brain*. Grosset/Putnam Books.
- [der Linden et al., 1999] der Linden, M. V., Seron, X., Gall, D. L., and Andrès, P. (1999). *Neuropsychologie des lobes frontaux*. Solal.
- [El-Nasr et al., 2000] El-Nasr, M., Ioerger, T., and Yen, J. (2000). A web of emotions. In *Proc. of the Workshop on Emotion-Based Agent Architectures (EBAA'99) at the 3rd International Conference on Autonomous Agents (Agents'99)*.
- [Florea and Kalisz, 2003] Florea, A. and Kalisz, E. (2003). Behavior anticipation based on beliefs, desires and emotions. In *Proc. of the 6th International Conference on Computing Anticipatory Systems (CASYS 03)*.
- [Gras et al., 2003] Gras, R., Peter, P., Baquedano, S., and Philippé, J. (2003). Structuration de comportements de réponse à un questionnaire par des méthodes multi-dimensionnelles. *Extraction et Gestion des Connaissances, Extraction des Connaissances et Apprentissage*, 17(1–3) :105–118.
- [Kapoor et al., 2001] Kapoor, A., Mota, S., and Picard, R. (2001). Towards a learning companion that recognizes affect. In *Proc. of Emotional and Intelligent : The Tangled knot of social cognition, AAAI Fall Symposium*.
- [OMG, 2003] OMG (2003). Unified modeling language specification, version 1.5.
- [Orthonoy et al., 1988] Orthonoy, A., Clore, G., and Collins, A. (1988). *The cognitive structure of emotions*. Cambridge University Press.
- [Rao and Georgeff, 1991] Rao, A. and Georgeff, M. (1991). Modeling rational agents within a bdi-architecture. In Allen, J., Fikes, R., and Sandewal, E., editors, *Proc. of the 2nd International Conference on Principles of Knowledge Representation and Reasoning (KR'91)*, pages 473–484. Morgan Kaufmann publishers Inc.
- [Rickel et al., 2002] Rickel, J., Marsella, S., Gratch, J., Hill, R., and Traum, D. (2002). Steve goes to bosnia : Towards a new generation of virtual humans for interactive experience. *IEEE Intelligent Systems*, 17(4) :32–38.
- [Sandoval, 1997] Sandoval, V. (1997). *L'informatique décisionnelle*. Hermès Sciences.
- [Wooldridge, 2002] Wooldridge, M. (2002). *An Introduction to MultiAgent Systems*. John Wiley & Sons.